Limits of Probabilistic Safety Guarantees when Considering Human Uncertainty

Richard Cheng[1], Richard M. Murray[1], and Joel W. Burdick[1]

***Abstract*— When autonomous robots interact with humans, such as during 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)* current learning-based methods are ill-equipped to compute *accurate* confidence bounds at such low δ. Using human driving data (from the *highD* dataset), as well as synthetically generated data, we show that current uncertainty models use inaccurate distributional assumptions to describe human behavior and/or require infeasible amounts of data to accurately learn confidence bounds for $\delta \leq 10^{-8}$. These two issues result in unreliable confidence bounds, which can have dangerous implications if deployed on safety-critical systems.**

I. INTRODUCTION

Autonomous robots will be increasingly deployed in unstructured human environments (e.g. roads and malls) where they must safely carry out tasks in the presence of other moving 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]–[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 aim to 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 applications require δ that are orders of magnitude lower [6].

> *Suppose a robot/car is guaranteed safe with probability $1-\delta$ across every $10s$ planning horizon. Given $\delta \approx 0.001$, we could expect a safety violation every 3 hrs. For reference, based on NHTSA data [7], human drivers have an effective safety threshold $\delta < 10^{-7}$.*

[1]All authors are with the California Institute of Technology, Pasadena, CA, USA rcheng@caltech.edu, murray@cds.caltech.edu, jwb@robotics.caltech.edu

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 data of human driving from the `highD` dataset [8]. We found that *even under extremely generous assumptions*, learned models are highly inaccurate in capturing human behavior at low δ, often mispredicting the probability of rare events by several orders of magnitude. Furthermore, we show that increasing dataset sizes will not sufficiently improve accuracy of learned uncertainty models.

Our results highlight potential 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 crucial problem (rather than providing a solution), we conclude by discussing alternative approaches that can address these limitations by combining *(a)* learned patterns of behavior and *(b)* prior knowledge encoding human interaction rules.

Before proceeding, we emphasize three critical points regarding our results:

- We focus on in-distribution error, rather than out-of-distribution error. I.e., we highlight the fundamental inability of uncertainty models to accurately capture distributions at very low δ, regardless of generalization.
- We focus *not* on robust control algorithms, but rather on the learned uncertainties that such algorithms leverage.
- We distinguish *motion predictors* from *uncertainty models*. While recent performance of motion predictors has drastically improved [9], they all leverage an underlying *uncertainty model* (see Table 1) to capture the probability of uncommon events. E.g. most neural network motion predictors output a Gaussian uncertain prediction. This paper focuses on errors associated with *uncertainty models* (which propagate to the motion predictors).

II. 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 learned distribution). These uncertainty models capture noise and the effects

Uncertainty Model Class	Example Works	Min. Safety Threshold
Gaussian Process	[2], [10]–[12]	$\delta \geq 0.001$
Dynamics w/ Gaussian Noise	[13]–[15]	$\delta \geq 0.001$
Bayesian NN	[16]–[18]	$\delta \geq 0.05$
Noisy Rational Model	[3]	$\delta \geq 0.01$
Hidden Markov Model / Markov Chain	[19], [20]	$\delta \geq 0.01$
Quantile Regression	[5]	$\delta \geq 0.05$
Scenario Optimization	[21]–[23]	$\delta \geq 0.01$
Generative Models (e.g. GANs)	[9], [24], [25]	N/A

TABLE I: 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.

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 [2], [11], [12], [14], [26]. There are several extensions, such as the IGP model [27] (which accounts for interaction between multiple agents), or others [28], [29]. 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) [15], [30].
- **Quantile Regression:** This approach computes quantile bounds over the trajectories of other agents at a given confidence level, δ. This approach benefits from not assuming an uncertainty distribution over trajectories [5], [31].
- **Scenario Optimization:** This approach computes a predicted set over other agents' actions based on samples of previously observed scenarios [32]. It is distribution-free (i.e. does not assume a parametric uncertainty distribution) [21]–[23], [33]. [34], [35] do not use scenario optimization, but their work based on computing minimum support sets follows a similar flavor.
- **Noisy (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 Eq. (5). The uncertainty in the action is captured by this distribution, which relies on an accurate model of the human's value function [1], [3], [36]–[38].
- **Generative Models (CVAE, GAN):** These models learn an implicit distribution over trajectories. Rather than give an explicit distribution, they generate random trajectories that try to model the true distribution [9], [24], [25].
- **Hidden Markov Model (HMM) / Markov Chain:** These models 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, $1-\delta$ [19], [20], [39]–[43].
- **Uncertainty Quantifying (UQ) Neural Networks:** These approaches do not constitute a separate class of uncertainty models, but refer to methods that train a neural network to capture the distribution over other agents' trajectories [16], [18], [44]. 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 [45], deep ensembles [46], Monte-Carlo dropout [47]). These models can also quantify uncertainty over discrete states (i.e. infer the hidden state in HMMs) [48], [49].

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 do *not* focus on these mechanisms (i.e. robust control algorithms) for guaranteeing safety; rather we focus on the issue of learning/modeling trajectory uncertainty, which such mechanisms must leverage for their safety guarantees.

III. 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.

We highlight these limitations by testing prevalent modeling assumptions on real-world driving data from the highD driving dataset [8], which captures vehicles driving on German highways. From this dataset, we extract all trajectories of length 10 seconds, $\tau_{[0,10]}$, along with their corresponding environmental context, $\mathcal{E}_\tau$ (i.e. position/velocity of surrounding cars). We then split the trajectories into a training set, $(\tau_{[0,10]}^{(train)}, \mathcal{E}_\tau^{(train)}) \in \mathcal{D}^{train}$, and test set, $(\tau_{[0:10]}^{(test)}, \mathcal{E}_\tau^{(test)}) \in$

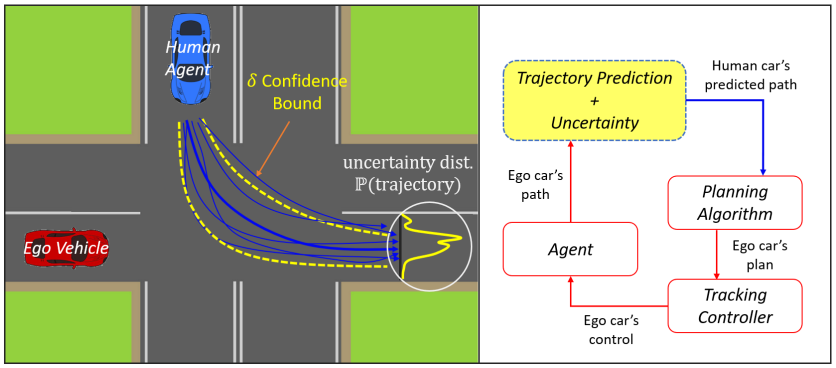


Fig. 1: **(Left)** In this example the red car must take into account the blue car's trajectory – and its uncertainty – in its plan to progress safely through the intersection. The dashed yellow curves denote the boundary of a tube that defines the δ confidence bound over trajectories. The white circle 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*.

$\mathcal{D}^{test}$. For every test trajectory, $(\tau^{(test)}_{[0:10]}, \mathcal{E}^{(test)}_{\tau}) \in \mathcal{D}^{test}$, we collect all trajectories in the training set in equivalent scenarios,

$$\mathcal{M}(\tau^{(test)}_{[0:10]}, \mathcal{E}^{(test)}_{\tau}) = \Big\{ \tau_{[0:10]} \;\Big|\; (\tau_{[0:10]}, \mathcal{E}_{\tau}) \in \mathcal{D}^{train} \;, \\ \|\tau_{[0:2]} - \tau^{(test)}_{[0:2]}\|_{\infty} < \epsilon \quad , \quad \|\mathcal{E}_{\tau} - \mathcal{E}^{(test)}_{\tau}\|_{\infty} < \epsilon_{\mathcal{E}} \Big\} \;. \tag{1}$$

We define *equivalent scenarios* as the set of trajectories with similar environmental context that are ϵ-close ($\epsilon = 2$ft) over their first $2s$. Therefore, $\mathcal{M}(\tau^{(test)}, \mathcal{E}^{(test)}_{\tau})$ denotes the set of scenarios (within the training set, $\mathcal{D}^{train}$) that are equivalent to $(\tau^{(test)}, \mathcal{E}^{(test)}_{\tau})$. We chose a past observation horizon of $2s$ following [49], but found that the observed trends did not change considerably when using $1s$ or $3s$ for the past observation horizon.

For every test trajectory, $(\tau^{(test)}, \mathcal{E}^{(test)}_{\tau}) \in \mathcal{D}^{test}$, we fit optimal parameters for an uncertainty model (e.g. Gaussian) to the equivalent training scenarios, $\mathcal{M}(\tau^{(test)}, \mathcal{E}^{(test)}_{\tau})$, 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 the 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 $x = (\tau_{[0:2]}, \mathcal{E}_{\tau})$, and 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,

$$L^{out} = \mathbb{E}\big[m\big(\mathcal{A}(x) \| \hat{F}(x)\big)\big] \;, \tag{2}$$

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,

$$L^{unseen} = \frac{1}{N_{unseen}} \sum_{x \in D^{test}} \big[m\big(\hat{\mathcal{A}}(x) \| \hat{F}(x)\big)\big] \;, \\ L^{seen} = \frac{1}{N_{seen}} \sum_{x \in D^{test} \cap \mathcal{D}^{train}} \big[m\big(\hat{\mathcal{A}}(x) \| \hat{F}(x)\big)\big] \;, \tag{3}$$

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 did not train 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,

$$L^{out} \underbrace{\geq}_{\text{distribution gap}} L^{unseen} \underbrace{\geq}_{\text{generalization gap}} L^{seen}. \tag{4}$$

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 replanning: Most motion planning algorithms re-plan their trajectory at some fixed frequency (e.g. 1Hz). To account for this, we examine prediction error (e.g. violation of the $\delta-$uncertainty bound) only within a short re-planning horizon. I.e. the prediction must only be accurate within this replanning horizon. The horizon is set to 2 sec.

Incorporating conservative assumptions: To further highlight the fundamental limitations of learning uncertainty models of human behavior, since many prediction algorithms leverage goal inference, we assume that an oracle gives us the target lane of every trajectory. Note that our aim 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.

Summarizing, we consider *(a)* there is no distribution gap, *(b)* there is no generalization gap, and *(c)* we are given the target lane of every trajectory.

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

A. Gaussian Uncertainty Models

We start by analyzing the popular Gaussian uncertainty model, used in most UQ neural networks [18], Gaussian process models [2], and robust regression [4], [29]. These approaches model the data and its uncertainty with a Gaussian distribution (see top 3 rows in Table 1).

Using the procedure outlined at the beginning of Section III, 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. minimizes L^{seen}). Figure 2 ($K = 1$) 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 (within a 2s re-planning horizon) for a specified δ. If the data followed a perfect Gaussian distribution, each curve in Fig. 2 would track the dotted black line (i.e. keep a ratio near 1). However, we see that while the Gaussian model might be valid for $\delta \geq 0.01$, it is highly inaccurate outside this range, posing a problem for safety-critical applications.

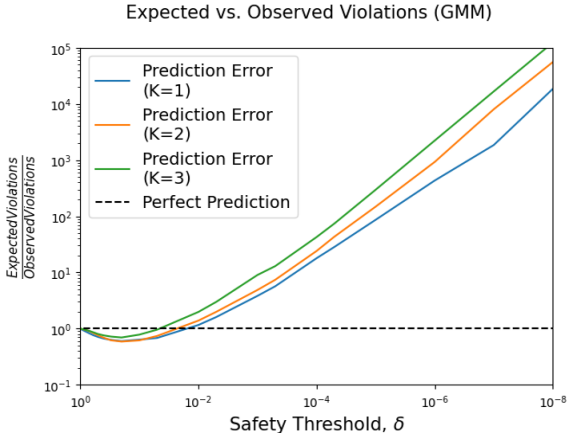


Fig. 2: Prediction error vs. safety threshold, δ, using Gaussian mixture models on the highD dataset, considering a $2s$ re-planning horizon. K denotes the number of mixtures used, with $K = 1$ denoting a standard Gaussian distribution. The dashed black line represents a perfect prediction model.

Gaussian mixture models (GMM): One might point out that problems with the Gaussian model could be alleviated using 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 [27]. However, when we trained GMMs on the same data with different numbers of mixtures ($K = 2, ..., 4$), prediction performance on test data did not improve for low δ (see Fig. 2). These results illustrate limitations of any Gaussian-based uncertainty model (IGP, GMM, etc.), by highlighting that human behavioral variation is inherently non-Gaussian.

In addition to the issue of inaccurate distributional assumptions, the confidence bounds at level $\delta \approx 10^{-8}$ become very large, making planning around these bounds difficult or potentially infeasible. Figure 3 shows the 5σ confidence tube projecting the position of a car forward in time, based on the training data. Note that the 5σ ellipsoid (corresponding to $\delta < 10^{-6}$) encroaches on each lane, making it difficult for other cars to drive alongside it. This is because, although the car will typically stay in its lane, in rare instances (as shown in Figure 3) it will unexpectedly swerve into the other lane. This illustrates the difficulty of balancing the safety-efficiency tradeoff, as accounting for very rare events may be necessary for safety-critical applications, but this also introduces extreme conservatism.

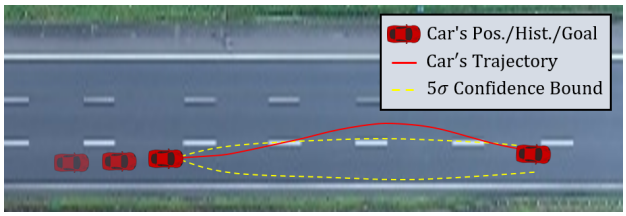


Fig. 3: Example of a car's trajectory, along with the approximate 5σ confidence bound computed from the training trajectories, given the car's target lane 8 seconds in the future.

To further emphasize fragility of the Gaussian model at low δ, we generated synthetic 2D data from different, known distributions, and 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. Details and results are in Appendix A (found at [50]).

B. 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 [37], 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, which places a strong assumption on the shape of the uncertainty distribution and assumes that there is a single "optimal" trajectory:

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

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 target lane, optimal data fit, no generalization gap – these models are ill-equipped to provide safety guarantees for safety-critical systems (e.g. $\delta < 10^{-8}$).

C. 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 highD dataset 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 [51].

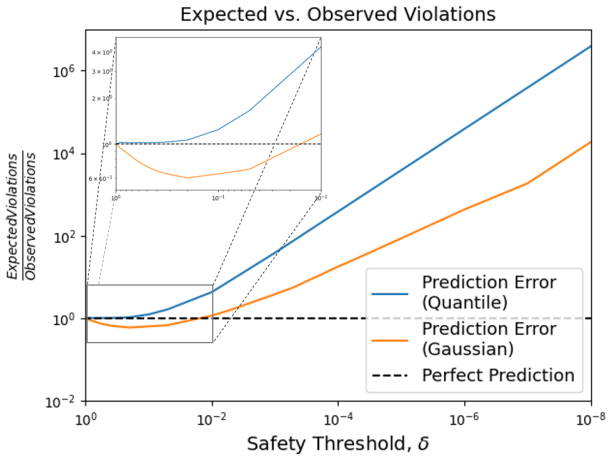


Fig. 4: Prediction error vs. safety threshold, δ using computed quantile bounds or Gaussian uncertainty model on the highD dataset (assuming $2s$ re-planning horizon). The dashed black line represents a perfect prediction model.

Figure 4 shows the ratio of the observed to expected number of test trajectories outside each quantile at safety threshold δ. As seen in the plot, the quantile regression model performs much better than the Gaussian model for $\delta > 0.1$. However, performance rapidly deteriorates as δ decreases, making estimated confidence bounds meaningless, since they fail to predict violation probabilities.

This result 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.

Improving Accuracy with Increasing Data: Given the availability of increasingly large robotics datasets, we should ask whether 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,

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

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 size of our training set, we capture how δ_{min} varies with the amount of training data, shown in Fig. 5a. The trend shown in Fig. 5a is surprisingly linear ($r^2 = 0.995$), which held across different sections of the dataset (i.e. different highways). This scaling is consistent with the lower bound on sample complexity derived in [52], shown in Eq. (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 5b shows that we would need trillions of kilometers of driving data to achieve accurate quantile bounds, even under extremely generous assumptions (e.g. perfect generalization). For reference, in 2018, approximately 5 trillion kilometers were driven in total across *all* cars/trucks in the U.S. [7].

We conducted the same analysis on synthetic 2D data, and found the same trends seen in Figures 4 and 5. Details and results are in Appendix B (found at [50]).

Quantile Regression as a Fundamental Limitation: One might be tempted to conclude from Figure 5b that we should look for alternative methods (to quantile regression) that have better data efficiency / sample complexity. Note that all methods providing confidence bounds over trajectories at a specified safety threshold δ 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 [52]: To guarantee $Pr(\text{error}) \leq \delta$, we require

$$N(\delta, M) = \Omega\Big(\frac{1}{\delta}\ln\big(\frac{1}{\delta}\big) + \frac{\text{VCdim}(M)}{\delta}\Big) \tag{7}$$

where VCdim(M) is the VC dimension of the utilized model M (see [52] for proof). The linear trend in Fig. 5 (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 analysis suggests that alternative methods cannot provide confidence bounds with better data scaling than shown in Figure 5.*

D. Other Uncertainty Models

Due to space constraints, we discuss remaining models in Appendix C (found at [50]), including: *(1)* generative models (e.g. CVAE, GAN), *(2)* scenario optimization, and *(3)*

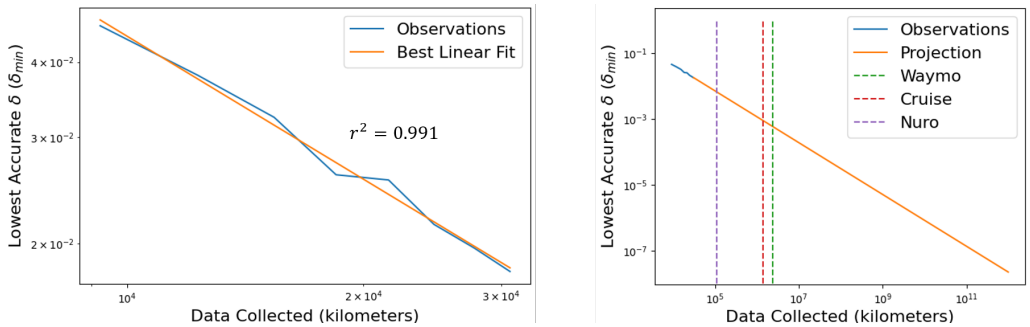


Fig. 5: *(a)* Smallest accurate δ versus amount of data collected. The trend is highly linear ($r^2 \approx 0.995$), and holds across different sections of the dataset. *(b)* Projection showing the expected amount of data required to obtain an accurate safety threshold δ_{min}. The dashed lines show the number of kilometers driven in California in 2019 by Waymo, Cruise, and Nuro.

hidden Markov models (HMM). However, below, we briefly describe fundamental problems each of 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 [53]. Also, using state-of-the-art models [9], [24], [25], it would require *at least* a day to generate enough trajectories to certify safety with $\delta \approx 10^{-8}$.
- HMMs are distinct as they learn probabilities over discrete states (e.g. goal positions). However, we show that even with a known observation function, $\mathbb{P}(\text{obs}|\text{state})$, it is highly unlikely to obtain sufficient confidence ($\delta \approx 10^{-8}$) of being in a given state.

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). Our results highlight *best-case* performance bounds for each class of model uncertainties, given optimal fit to the data. Thus, using neural networks to parameterize the model uncertainty will only yield worse performance.

IV. CONCLUSION AND FUTURE WORK

Our main message is that *even under extremely generous assumptions*, current models of human uncertainty are unable to extend safety guarantees to the confidence levels, e.g. $\delta < 10^{-8}$, that are needed for widespread adoption of safety-critical autonomy in human environments.

Learned uncertainty distributions become highly inaccurate at low δ, undermining any claimed guarantees of safety..

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 a million samples to *observe* a one-in-a-million event, and we need many more samples to *reliably predict* those events. While it is theoretically possible that huge datasets could eventually enable accurate prediction of rare events, our analysis shows that such amounts of data are far from feasible in the near future (even ignoring generalization issues and computational cost).

Human 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, 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 position with confidence $1-10^{-8}$. 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. However, this could be a promising avenue of research: to simultaneously learn multiple predictors [54], *while enforcing 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) [55], which bound uncertainty in useful ways. These rules can be encoded through *assume-guarantee contracts* [56]. 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 it. We thus propose trading one challenge for another: rather than learning uncertainty bounds that agents obey with probability $1-\delta$, we should instead aim to specify interpretable contracts (i.e. behavioral constraints) with learned components that agents must surely obey.

We believe such a framework is necessary to move away from treating uncertainty in human behavior purely as a random process. Instead, human uncertainty can be constrained by combining *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," in *Robotics: Science and Systems*, 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*, vol. 35, no. 1, pp. 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," *International Journal of Robotics Research*, vol. 39, no. 2-3, pp. 250–265, March 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. Agha-mohammadi, and E. A. Theodorou, "Deep learning tubes for tube mpc," *arXiv*, 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. Kloeker, 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] A. Gupta, J. Johnson, L. Fei-Fei, S. Savarese, and A. Alahi, "Social gan: Socially acceptable trajectories with generative adversarial networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[10] J. F. Fisac, E. Bronstein, E. Stefansson, D. Sadigh, S. S. Sastry, and A. D. Dragan, "Hierarchical game-theoretic planning for autonomous vehicles," in *International Conference on Robotics and Automation (ICRA)*, 2019, pp. 9590–9596.

[11] A. Hakobyan and I. Yang, "Learning-Based Distributionally Robust Motion Control with Gaussian Processes," *arXiv*, 2020.

[12] 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.

[13] W. Xu, J. Pan, J. Wei, and J. M. Dolan, "Motion planning under uncertainty for on-road autonomous driving," in *IEEE ICRA*, 2014.

[14] D. Sadigh and A. Kapoor, "Safe control under uncertainty with Probabilistic Signal Temporal Logic," in *Robotics: Science and Systems*, vol. 12, 2016.

[15] M. Forghani, J. M. McNew, D. Hoehener, and D. Del Vecchio, "Design of driver-assist systems under probabilistic safety specifications near stop signs," *IEEE Transactions on Automation Science and Engineering*, vol. 13, no. 1, pp. 43–53, 2016.

[16] G. Kahn, A. Villaflor, V. Pong, P. Abbeel, and S. Levine, "Uncertainty-Aware Reinforcement Learning for Collision Avoidance," *arXiv*, 2017.

[17] 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," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2020, pp. 4093–4099.

[18] 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," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2020, pp. 7344–7350.

[19] 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*, 2014, pp. 56–61.

[20] 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 *2015 IEEE Intelligent Vehicles Symposium (IV)*, 2015, pp. 1126–1133.

[21] G. Cesari, G. Schildbach, A. Carvalho, and F. Borrelli, "Scenario model predictive control for lane change assistance and autonomous driving on highways," *IEEE Intelligent Transportation Systems Magazine*, vol. 9, no. 3, pp. 23–35, 2017.

[22] Y. Chen, S. Dathathri, T. Phan-Minh, and R. M. Murray, "Counter-example guided learning of bounds on environment behavior," in *Proceedings of the Conference on Robot Learning*, vol. 100. PMLR, Nov 2020, pp. 898–909.

[23] H. Sartipizadeh and B. Açıkmeşe, "Approximate convex hull based scenario truncation for chance constrained trajectory optimization," *Automatica*, vol. 112, 2020.

[24] 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/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 1349–1358.

[25] T. Salzmann, B. Ivanovic, P. Chakravarty, and M. Pavone, "Trajectron++: Multi-Agent Generative Trajectory Forecasting With Heterogeneous Data for Control," *arXiv*, 2020.

[26] D. Ellis, E. Sommerlade, and I. Reid, "Modelling pedestrian trajectory patterns with gaussian processes," in *IEEE 12th International Conference on Computer Vision Workshops, ICCV Workshops*, 2009, pp. 1229–1234.

[27] 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," *The International Journal of Robotics Research*, vol. 34, no. 3, pp. 335–356, 2015.

[28] 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 *Algorithmic Foundations of Robotics XI: Selected Contributions of the Eleventh International Workshop on the Algorithmic Foundations of Robotics*. Springer International Publishing, 2015, pp. 161–177.

[29] A. Liu, G. Shi, S.-J. Chung, A. Anandkumar, and Y. Yue, "Robust Regression for Safe Exploration in Control," vol. 120. PMLR, Jun 2020, pp. 608–619.

[30] 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 *2013 IEEE Intelligent Vehicles Symposium (IV)*, 2013, pp. 2329–2334.

[31] N. Tagasovska and D. Lopez-Paz, "Single-Model Uncertainties for Deep Learning," in *Neural Information Processing Systems*, 2018.

[32] M. C. Campi and S. Garatti, "Wait-and-judge scenario optimization," *Mathematical Programming*, 2018.

[33] A. Carvalho, S. Lefévre, G. Schildbach, J. Kong, and F. Borrelli, "Automated driving: The role of forecasts and uncertainty - A control perspective," in *European Journal of Control*, vol. 24, 2015, pp. 14–32.

[34] K. Driggs-Campbell, V. Govindarajan, and R. Bajcsy, "Integrating Intuitive Driver Models in Autonomous Planning for Interactive Maneuvers," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 12, pp. 3461–3472, 2017.

[35] K. Driggs-Campbell, R. Dong, and R. Bajcsy, "Robust, informative human-in-the-loop predictions via empirical reachable sets," *IEEE Transactions on Intelligent Vehicles*, vol. 3, no. 3, pp. 300–309, 2018.

[36] 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.

[37] 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.

[38] 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," in *Proceedings of the 2020 ACM/IEEE International Conference on Human-Robot Interaction*. Association for Computing Machinery, 2020, p. 43–52.

[39] 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 *Proceedings of the 3rd ACM/IEEE International Conference on Human Robot Interaction*. Association for Computing Machinery, 2008, p. 367–374.

[40] C. L. McGhan, A. Nasir, and E. M. Atkins, "Human intent prediction using Markov decision processes," *Journal of Aerospace Information Systems*, 2015.

[41] 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.

[42] C. P. Lam and S. S. Sastry, "A POMDP framework for human-in-the-loop system," in *IEEE Conference on Decision and Control*, 2014.

[43] D. Tran, W. Sheng, L. Liu, and M. Liu, "A Hidden Markov Model based driver intention prediction system," in *IEEE International Conference on Cyber Technology in Automation, Control, and Intelligent Systems (CYBER)*, 2015, pp. 115–120.

[44] T. Gindele, S. Brechtel, and R. Dillmann, "A probabilistic model for estimating driver behaviors and vehicle trajectories in traffic environments," in *13th International IEEE Conference on Intelligent Transportation Systems*, 2010, pp. 1625–1631.
[45] C. Blundell, J. Cornebise, K. Kavukcuoglu, and D. Wierstra, "Weight uncertainty in neural networks," in *International Conference on Machine Learning*, vol. 37. PMLR, Jul 2015, pp. 1613–1622.
[46] B. Lakshminarayanan, A. Pritzel, and C. Blundell, "Simple and scalable predictive uncertainty estimation using deep ensembles," in *Advances in Neural Information Processing Systems 30*, 2017, pp. 6402–6413.
[47] Y. Gal and Z. Ghahramani, "Dropout as a bayesian approximation: Representing model uncertainty in deep learning," in *International Conference on Machine Learning*, 2016.
[48] Y. Hu, W. Zhan, and M. Tomizuka, "Probabilistic Prediction of Vehicle Semantic Intention and Motion," in *IEEE Intelligent Vehicles Symposium*, vol. 2018-June, 2018, pp. 307–313.
[49] W. Ding, J. Chen, and S. Shen, "Predicting vehicle behaviors over an extended horizon using behavior interaction network," in *International Conference on Robotics and Automation (ICRA)*, 2019, pp. 8634–8640.
[50] [Online]. Available: https://rcheng805.github.io/
[51] K. Pelckmans, J. D. Brabanter, J. A.K. Suykens, and B. De Moor, "Support and Quantile Tubes," *arXiv*, 2008.
[52] 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.
[53] S. Arora, A. Risteski, and Y. Zhang, "Do GANs learn the distribution? some theory and empirics," in *International Conference on Learning Representations*, 2018.
[54] A. Filos, P. Tigas, R. McAllister, N. Rhinehart, S. Levine, and Y. Gal, "Can Autonomous Vehicles Identify, Recover From, and Adapt to Distribution Shifts?" in *International Conference on Machine Learning*, 2020.
[55] M. E. Bratman, "Shared Cooperative Activity," *The Philosophical Review*, 1992.
[56] T. Phan-Minh, K. X. Cai, and R. M. Murray, "Towards assume-guarantee profiles for autonomous vehicles," in *IEEE 58th Conference on Decision and Control (CDC)*, 2019, pp. 2788–2795.

APPENDIX 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 3A. Each 2D data point is analogous to a trajectory, $a = \tau_{[2:10]} \sim \mathcal{A}$, in the highD driving dataset. Therefore, the goal is still to learn the model $\hat{F}$ that best matches the data distribution $\mathcal{A}$, minimizing $m(\mathcal{A}||\hat{\mathcal{F}})$. 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.

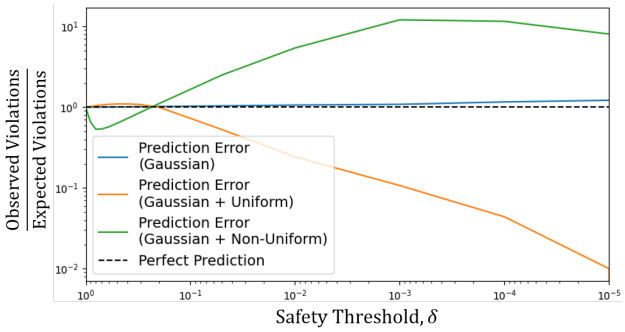


Fig. 6: Prediction error vs. safety threshold, δ, using a Gaussian uncertainty model on synthetic 2D data generated from 3 different distributions. The dashed black line represents a perfect prediction model. Significant prediction error arises when the underlying data distribution is non-Gaussian.

Figure 6 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 3A by illustrating that significant prediction error inevitably arises, regardless of the amount of training data, when the underlying data distribution is non-Gaussian.

APPENDIX B: QUANTILE REGRESSION (SYNTHETIC DATA)

We repeated the quantile regression experiments from Section 3C, 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 3C (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.

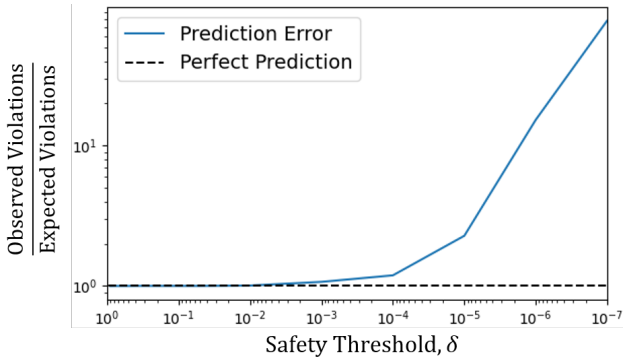


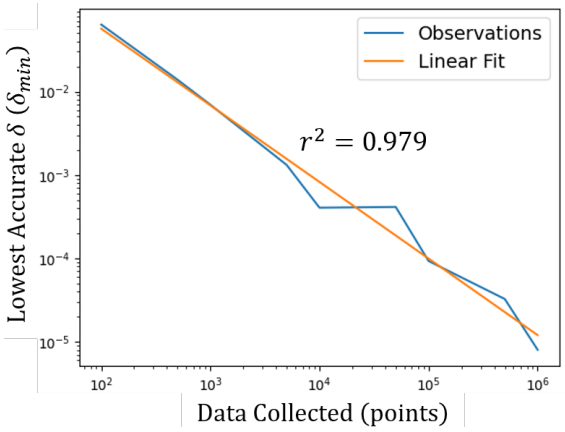
Fig. 7: Prediction error vs. safety threshold, δ under computed quantile bounds on synthetic 2D data. The dashed black line represents a perfect prediction model.

Figure 7 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 3C.

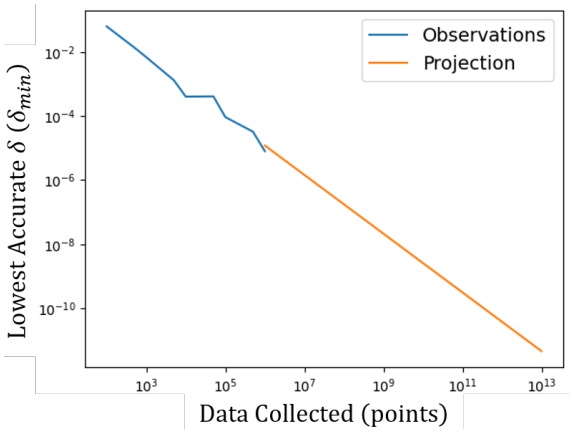
Using the synthetic data, we computed the smallest accurate safety threshold, δ_{min}, as a function of the amount of training data, N. This threshold δ_{min} was defined as follows,

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

where we set $\varepsilon = 0.5$, which represents the vertical distance between the blue curve in Fig 7 and the dotted black line. Therefore, δ_{min} represents the smallest δ such that our computed quantile bounds are ε-accurate (as described in



(a)



(b)

Fig. 8: *(a)* Smallest accurate δ versus amount of data using synthetic 2D data. The trend is highly linear ($r^2 = 0.979$), *(b)* Projection showing the expected amount of data required to obtain an accurate safety threshold δ_{min}.

Section 3C). Figure 8(a) 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 8(b) shows the extrapolation of this trend towards lower δ_{min}. This result reinforces the point made in Section 3C that quantile regression can be very accurate for larger δ, but it may not be feasible to collect enough data to reach safety thresholds $\delta_{min} \leq 10^{-8}$.

APPENDIX C: OTHER UNCERTAINTY MODELS

A. 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 [9], or $\approx 0.001s$ with Trajectron [25]. 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 [53]. Furthermore, the theoretical data efficiency bound described by Eq. (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.

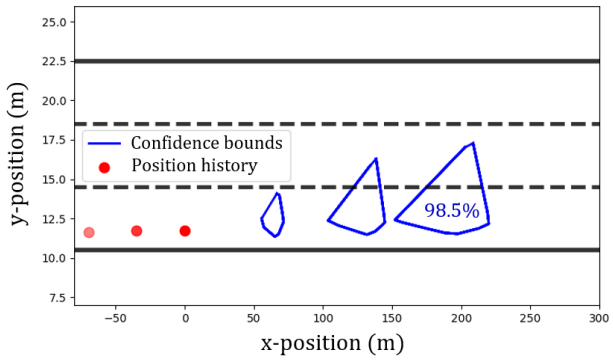
B. Scenario Optimization Model

Scenario Optimization is an appealing approach because (like quantile regression) it does not assume an underlying distribution over the data [23]. 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*.

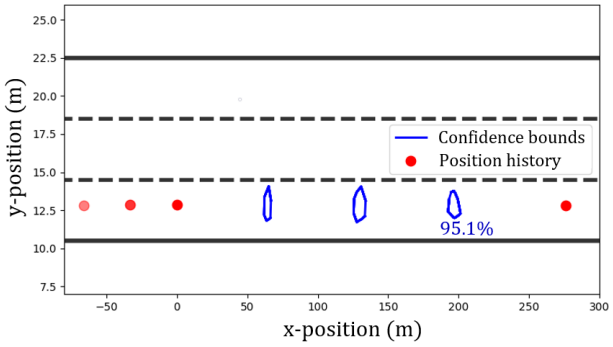
With this approach, the safety threshold, δ, is a direct function of the amount of observed data [32]; in other words $\delta = \delta(N)$, where N is the number of training trajectories or "samples". Therefore, we cannot set arbitrarily small confidence levels (e.g. $\delta = 10^{-8}$). While this prevents 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 9 shows the predicted confidence bounds for two representative driving instances; in Fig. 9a, the goal position is *not* given but in in Fig. 9b, 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% (Fig. 9a) or 95.1% (Fig. 9b) probability.

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



(a)



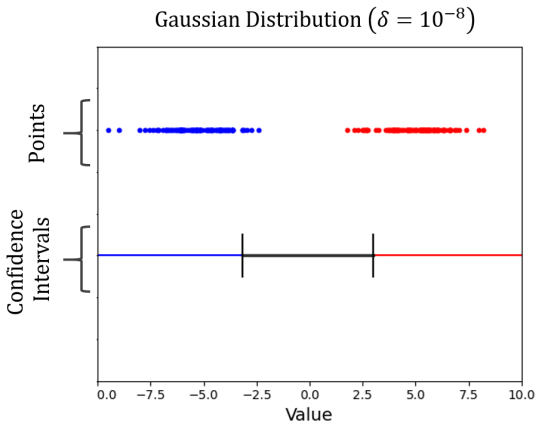
(b)

Fig. 9: 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. *(a)* 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. *(b)* The target 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.

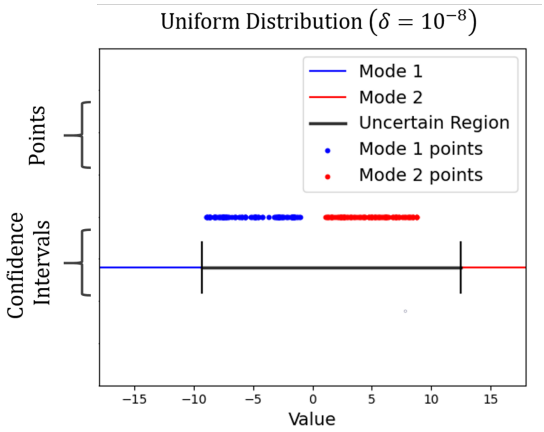
However, the safety threshold $\delta(N)$ was always large ($\delta \in [0.02, 0.41]$) and unable to be arbitrarily defined, which makes the scenario optimization approach currently inapplicable to many safety-critical applications. This is consistent with our conclusion in Section 3C, that much more data is necessary to obtain reliable, probabilistic bounds.

C. Hidden Markov Models

Rather than reasoning about uncertainty only over trajectories, many methods in the POMDP literature reason about uncertainty over discrete intentions. Most often, these discrete intentions denote different goal positions for the agent, but they could also denote different operational modes (e.g. yield vs. no yield). Hidden Markov models enable us to compute an agent's most likely intention, which proves useful in solving many challenging problems. However, when guaranteeing safety with safety threshold δ, intention must be correctly inferred with probability $1 - \delta$. Issues arise when the intention must be inferred with very high confidence $\delta \leq 10^{-8}$.



(a) Data for each mode generated from a Gaussian distribution.



(b) Data for each mode generated from a uniform distribution.

Fig. 10: 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).

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

$$\mathbb{P}(\text{mode} \mid x) = \frac{\mathbb{P}(x \mid \text{mode})\ \mathbb{P}(\text{mode})}{\mathbb{P}(x)}. \tag{9}$$

Figure 10 shows these intervals when the points were generated from either a Gaussian distribution, or a uniform distribution. The interval covered by the gray line denotes the interval in which we can *not* classify (with δ confidence)

a point's mode. We note that the gray line extends across a significant portion of the data range, but is reasonable when the underlying distribution of points in each mode is *perfectly Gaussian*. However, when the generated data is uniformly random, the uncertainty interval stretches across the entire range of data. This suggests that inferring intention or hidden "modes" under uncertainty will often be infeasible when considering very low safety thresholds, δ, especially since we have shown that human behavioral variation is non-Gaussian. Furthermore, we cannot compensate for this non-Gaussian variation as we do not have accurate knowledge of the true distribution.