Can Autonomous Vehicles Identify, Recover From, and Adapt to Distribution Shifts?

Angelos Filos* 1 Panagiotis Tigar* 1 Rowan McAllister 2 Nicholas Rhinehart 2 Sergey Levine 2 Yarin Gal 1

Abstract

Out-of-training-distribution (OOD) scenarios are a common challenge of learning agents at deployment, typically leading to arbitrary deductions and poorly-informed decisions. In principle, detection of and adaptation to OOD scenes can mitigate their adverse effects. In this paper, we highlight the limitations of current approaches to novel driving scenes and propose an epistemic uncertainty-aware planning method, called robust imitative planning (RIP). Our method can detect and recover from some distribution shifts, reducing the overconfident and catastrophic extrapolations in OOD scenes. If the model’s uncertainty is too great to suggest a safe course of action, the model can instead query the expert driver for feedback, enabling sample-efficient online adaptation, a variant of our method we term adaptive robust imitative planning (AdaRIP). Our methods outperform current state-of-the-art approaches in OOD scenes. Therefore, they can pick up non-causal features that generalise, perception-based, end-to-end approaches, which are evaluated (row-wise) by an ensemble of expert-likelihood models \(q_1, q_2, q_3\). Under models \(q_1\) and \(q_2\), the best plans are the catastrophic trajectories \(y^1\) and \(y^3\) respectively. Our epistemic uncertainty-aware robust (RIP) planning method aggregates the evaluations of the ensemble and proposes the safe plan \(y^3\). RIP considers the disagreement between the models and avoid overconfident but catastrophic extrapolations in OOD tasks.

1. Introduction

Autonomous agents hold the promise of systematizing decision-making to reduce catastrophes due to human mistakes. Recent advances in machine learning (ML) enable the deployment of such agents in challenging, real-world, safety-critical domains, such as autonomous driving (AD) regimes is concerning in life-critical applications, such as AD (Quionero-Candela et al., 2009; Leike et al., 2017).

Although there are relatively simple strategies (e.g., stay within the lane boundaries, avoid other cars and pedestrians) that generalise, perception-based, end-to-end approaches, while flexible, they are also susceptible to spurious correlations. Therefore, they can pick up non-causal features that lead to confusion in OOD scenes (de Haan et al., 2019).

Due to the complexity of the real-world and its ever-changing dynamics, the deployed agents inevitably face novel situations and should be able to cope with them, to at least (a) identify and ideally (b) recover from them, without failing catastrophically. These desiderata are not captured by the existing benchmarks (Ros et al., 2019; Codevilla et al., 2019) and as a consequence, are not satisfied by the...
Can Autonomous Vehicles Identify, Recover From, and Adapt to Distribution Shifts?

Figure 2. The robust imitative planning (RIP) framework. (a) Expert demonstrations. We assume access to observations x and expert state y pairs, collected either in simulation (Dosovitskiy et al., 2017) or in real-world (Caesar et al., 2019; Sun et al., 2019; Kesten et al., 2019). (b) Learning algorithm (cf. Section 3.1). We capture epistemic model uncertainty by training an ensemble of density estimators \{q(y|x; \theta_k)\}_{k=1}^K, via maximum likelihood. Other approximate Bayesian deep learning methods (Gal & Ghahramani, 2016) are also tested. (c) Planning paradigm (cf. Section 3.3). The epistemic uncertainty is taken into account at planning via the aggregation operator \(\oplus\) (e.g., \(\min_k\)), and the optimal plan \(y^*\) is calculated online with gradient-based optimization through the learned likelihood models.

2. Problem Setting and Notation

We consider sequential decision-making in safety-critical domains. A method is considered safe when it is accurate, with respect to some metric (cf. Sections 4, 6), and certain.

**Assumption 1** (Expert demonstrations). We assume access to a dataset, \(D = \{(x^i, y^i)\}_{i=1}^N\), of time-profiled expert trajectories (i.e., plans), \(y^i\), paired with high-dimensional observations, \(x^i\), of the corresponding scenes. The trajectories are drawn from the expert policy, \(y^i \sim \pi_{\text{expert}}(\cdot|\cdot)\).

Our goal is to approximate the (i.e., near-optimal) unknown expert policy, \(\pi_{\text{expert}}\), using imitation learning (Widrow & Smith, 1964; Pomerleau, 1989, IL), based only on the demonstrations, \(D\). For simplicity, we also make the following assumptions, common in the autonomous driving and robotics literature (Rhinehart et al., 2020; Du et al., 2019).

**Assumption 2** (Inverse dynamics). We assume access to an inverse dynamics model (Bellman, 2015, PID controller), \(I\), which performs the low-level control – inverse planning – \(a_t\) (i.e., steering, braking and throttling), provided the current and next states (i.e., positions), \(s_t\) and \(s_{t+1}\), respectively. Therefore, we can operate directly on state-only trajectories, \(y = (s_1, \ldots, s_T)\), where the actions are determined by the local planner, \(a_t = \mathcal{I}(s_t, s_{t+1})\), \(\forall t = 1, \ldots, T - 1\).

**Assumption 3** (Global planner). We assume access to a global navigation system that we can use to specify high-level goal locations \(G\) or/and commands \(C\) (e.g., turn left/right at the intersection, take the second exit).

**Assumption 4** (Perfect localization). We consider the provided locations (e.g., goal, ego-vehicle positions) as accurate, i.e., filtered by a localization system.

These are benign assumptions for many applications in robotics. If required, these quantities can also be learned from data, and are typically easier to learn than \(\pi_{\text{expert}}\).
3. Robust Imitative Planning

We seek an imitation learning method that (a) provides a distribution over expert plans; (b) quantifies epistemic uncertainty to allow for detection of OOD observations and (c) enables robustness to distribution shift with an explicit mechanism for recovery. Our method is shown in Figure 2. First, we present the model used for imitating the expert.

3.1. Bayesian Imitative Model

We perform context-conditioned density estimation of the distribution over future expert trajectories (i.e., plans), using a probabilistic “imitative” model \( q(y|x; \theta) \), trained via maximum likelihood estimation (MLE):

\[
\theta_{\text{MLE}} = \arg \max_{\theta} \mathbb{E}_{(x,y) \sim \mathcal{D}} [\log q(y|x; \theta)]. \tag{1}
\]

Contrary to existing methods in AD (Rhinehart et al., 2020; Chen et al., 2019), we place a prior distribution \( p(\theta) \) over possible model parameters \( \theta \), which induces a distribution over the density models \( q(y|x; \theta) \). After observing data \( \mathcal{D} \), the distribution over density models has a posterior \( p(\theta|\mathcal{D}) \).

**Practical implementation.** We use an autoregressive neural density estimator (Rhinehart et al., 2018), depicted in Figure 2b, as the imitative model, parametrised by learnable parameters \( \theta \). The likelihood of a plan \( y \) in context \( x \) to come from an expert (i.e., imitation prior) is given by:

\[
q(y|x; \theta) \triangleq \prod_{t=1}^{T} p(s_t | y_{<t}, x; \theta) = \prod_{t=1}^{T} \mathcal{N}(s_t; \mu(y_{<t}, x; \theta), \Sigma(y_{<t}, x; \theta)), \tag{2}
\]

where \( \mu(\cdot; \theta) \) and \( \Sigma(\cdot; \theta) \) are two heads of a recurrent neural network, with shared torso. We decompose the imitation prior as a telescopic product (cf. Eqn. (2)), where conditional densities are assumed normally distributed, and the distribution parameters are learned (cf. Eqn. (1)). Despite the unimodality of normal distributions, the autoregression (i.e., sequential sampling of normal distributions where the future samples depend on the past) allows to model multimodal distributions (Uria et al., 2016). Although more expressive alternatives exist, such as the mixture of density networks (Bishop, 1994) and normalising flows (Rezende & Mohamed, 2015), we empirically find Eqn. (2) sufficient.

The estimation of the posterior of the model parameters, \( p(\theta|\mathcal{D}) \), with exact inference is intractable for non-trivial models (Neal, 2012). We use ensembles of deep imitative models as a simple approximation to the posterior \( p(\theta|\mathcal{D}) \). We consider an ensemble of \( K \) components, using \( \theta_k \) to refer to the parameters of our \( k \)-th model \( q_k \), trained with via maximum likelihood (cf. Eqn. (1) and Figure 2b). However, any (approximate) inference method to recover the posterior \( p(\theta|\mathcal{D}) \) would be applicable. To that end, we also try Monte Carlo dropout (Gal & Ghahramani, 2016).

3.2. Detecting Distribution Shifts

The log-likelihood of a plan \( \log q(y|x; \theta) \) (i.e., imitation prior) is a proxy of the quality of a plan \( y \) in context \( x \) under model \( \theta \). We detect distribution shifts by looking at the disagreement of the qualities of a plan under models coming from the posterior, \( p(\theta|\mathcal{D}) \). We use the variance of the imitation prior with respect to the model posterior, i.e.,

\[
u(y) \triangleq \text{Var}_{p(\theta|\mathcal{D})} [\log q(y|x; \theta)] \tag{3}
\]

to quantify the model disagreement: Plans at in-distribution scenes have low variance, but high variance in OOD scenes. We can efficiently calculate Eqn. (3) when we use ensembles, or Monte Carlo, sampling-based methods for \( p(\theta|\mathcal{D}) \).

Having to commit to a decision, just the detection of distribution shifts via the quantification of epistemic uncertainty is insufficient for recovery. Next, we introduce an epistemic uncertainty-aware planning objective that allows for robustness to distribution shifts.

3.3. Planning Under Epistemic Uncertainty

We formulate planning to a goal location \( G \) under epistemic uncertainty, i.e., posterior over model parameters \( p(\theta|\mathcal{D}) \), as the optimization (Barber, 2012) of the generic objective, which we term robust imitative planning (RIP):

\[
y_{\text{RIP}}^G \triangleq \arg \max_y \mathbb{E}_{\theta \sim p(\theta|\mathcal{D})} \left[ \log p(y|G, x; \theta) \right] \tag{4}
\]

where \( \oplus \) is an operator (defined below) applied on the posterior \( p(\theta|\mathcal{D}) \) and the goal-likelihood is given, for example, by a Gaussian centred at the final goal location \( s_T^G \) and a pre-specified tolerance \( \epsilon \), \( p(G|y) = \mathcal{N}(y_T; s_T^G, \epsilon^2 I) \).

Intuitively, we choose the plan \( y_{\text{RIP}}^G \) that maximises the likelihood to have come from an expert demonstrator (i.e., “imitation prior”) and is “close” to the goal \( G \). The model posterior \( p(\theta|\mathcal{D}) \) represents our belief (uncertainty) about the true expert model, having observed data \( \mathcal{D} \) and from prior \( p(\theta) \) and the aggregation operator \( \oplus \) determines our level of awareness to uncertainty under a unified framework.

For example, a deep imitative model (Rhinehart et al., 2020) is a particular instance of the more general family of objectives described by Eqn. (4), where the operator \( \oplus \) selects a
single $\theta_k$ from the posterior (point estimate). However, this approach is oblivious to the epistemic uncertainty and prone to fail in unfamiliar scenes (cf. Section 4).

In contrast, we focus our attention on two aggregation operators due to their favourable properties, which take epistemic uncertainty into account: (a) one inspired by robust control (Wald, 1939) which encourages pessimism in the face of uncertainty and (b) one from Bayesian decision theory, which marginalises the epistemic uncertainty. Table 1 summarises the different operators considered in our experiments. Next, we motivate the used operators.

3.3.1. Worst Case Model (RIP-WCM)

In the face of (epistemic) uncertainty, robust control (Wald, 1939) suggests to act pessimistically – reason about the worst case scenario and optimise it. All models with non-zero posterior probability $p(\theta|D)$ are likely and hence our robust imitative planning with respect to the worst case model (RIP-WCM) objective acts with respect to the most pessimistic model, i.e.,

$$s_{\text{RIP-WCM}} \triangleq \arg \max_y \min_{\theta \in \text{supp}(p(\theta|D))} \log q(y|x; \theta). \quad (5)$$

The solution of the $\arg \max_y \min_{\theta}$ optimization problem in Eqn. (5) is generally not tractable, but our deep ensemble approximation enables us to solve it by evaluating the minimum over a finite number of $K$ models. The maximization over plans, $y$, is solved with online gradient-based adaptive optimization, specifically ADAM (Kingma & Ba, 2014). An alternative online planning method with a trajectory library (Liu & Atkeson, 2009) (c.f. Appendix D) is used too but its performance in OOD scenes is noticeably worse than online gradient descent.

Alternative, “softer” robust operators can be used instead of the minimum, including the Conditional Value at Risk (Embrechts et al., 2013; Rajeswaran et al., 2016, CVaR) that employs quantiles. CVaR may be more useful in cases of full support model posterior, where there may be a pessimistic but trivial model, for example, due to misspecification of the prior, $p(\theta)$, or due to the approximate inference procedure. Mean-variance optimization (Kahn et al., 2017; Kenton et al., 2019) can be also used, aiming to directly minimise the distribution shift metric, as defined in Eqn. (3).

Next, we present a different aggregator for epistemic uncertainty that is not as pessimistic as RIP-WCM and, as found empirically, works sufficiently well too.

3.3.2. Model Averaging (RIP-MA)

In the face of (epistemic) uncertainty, Bayesian decision theory (Barber, 2012) uses the predictive posterior (i.e., model averaging), which weights each model’s contribution according to its posterior probability, i.e.,

$$s_{\text{RIP-MA}} \triangleq \arg \max_y \int p(\theta|D) \log q(y|x; \theta) \, d\theta. \quad (6)$$

Despite the intractability of the exact integration, the ensemble approximation used allows us to efficiently estimate and optimise the objective. We call this method robust imitative planning with model averaging (RIP-MA), where the more likely models’ impacts are up-weighted according to the predictive posterior.

From a multi-objective optimization point of view, we can interpret the log-likelihood, $\log q(y|x; \theta)$, as the utility of a task $\theta$, with importance $p(\theta|D)$, given by the posterior density. Then RIP-MA in Eqn. (6) gives the Pareto efficient solution (Barber, 2012) for the tasks $\theta \in \text{supp}(p(\theta|D))$.

Table 1. Robust imitative planning (RIP) unified framework. The different aggregation operators applied on the posterior distribution $p(\theta|D)$, approximated with the deep ensemble (Lakshminarayanan et al., 2017) components $\theta_k$.

<table>
<thead>
<tr>
<th>Methods</th>
<th>Operator $\Theta$</th>
<th>Interpretation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Imitative Models</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Best Case (RIP-BCM)</td>
<td>$\max_{k} \log q_k$</td>
<td>Sample</td>
</tr>
<tr>
<td>Worst Case (RIP-WCM)</td>
<td>$\min_{k} \log q_k$</td>
<td>Max</td>
</tr>
</tbody>
</table>

**Figure 3.** RIP’s (ours) robustness to OOD scenarios, compared to (Codevilla et al., 2018, CIL) and (Rhinehart et al., 2020, DIM).

### 4. Benchmarking Robustness to Novelty

We designed our experiments to answer the following questions: Q1. Can autonomous driving, imitation-learning, epistemic-uncertainty unaware methods detect distribution shifts? Q2. How robust are these methods under distribution shifts, i.e., can they recover? Q3. Does RIP’s epistemic uncertainty quantification enable identification of novel scenes? Q4. Does RIP’s explicit mechanism for recovery from distribution shifts lead to improved performance? To that end, we conduct experiments both on real data, in
Section 4.1, and on simulated scenarios, in Section 4.2, comparing our method (RIP) against current state-of-the-art driving methods.

4.1. nuScenes

We first compare our robust planning objectives (cf. Eqn. (5–6)) against existing state-of-the-art imitation learning methods in a prediction task (Phan-Minh et al., 2019), based on nuScenes (Caesar et al., 2019), the public, real-world, large-scale dataset for autonomous driving. Since we do not have control over the scenes split, we cannot guarantee that the evaluation is under distribution shifts, but only test out-of-sample performance, addressing question Q4.

4.1.1. Metrics

For fair comparison with the baselines, we use the metrics from the ICRA 2020 nuScenes prediction challenge.

Displacement error. The quality of a plan, $y$, with respect to the ground truth prediction, $y^*$ is measured by the average displacement error, i.e.,

$$\text{ADE}(y) \triangleq \frac{1}{T} \sum_{t=1}^{T} \|s_t - s_t^*\|, \quad (7)$$

where $y = (s_1, \ldots, s_T)$. Stochastic models, such as our imitative model, $q(y|x; \theta)$, can be evaluated based on their samples, using the minimum (over $k$ samples) ADE (i.e., minADE$_k$), i.e.,

$$\text{minADE}_k(q) \triangleq \min_{y^1, \ldots, y^k} \text{ADE}(y^i). \quad (8)$$

In prior work, Phan-Minh et al. (2019) studied minADE$_k$ for $k > 1$ in order to assess the quality of the generated samples from a model, $q$. Although we report minADE$_k$ for $k = \{1, 5\}$, we are mostly interested in the decision-making (planning) task, where the driving agent commits to a single plan, $k = 1$. We also study the final displacement error (FDE), or equivalently minFDE$_1$, i.e.,

$$\minFDE_1(y) \triangleq \|s_T - s_T^*\|. \quad (9)$$

4.1.2. Baselines

We compare our contribution to state-of-the-art methods in the nuScenes dataset: the Multiple-Trajectory Prediction (Cui et al., 2019, MTP), MultiPath (Chai et al., 2019) and CoverNet (Phan-Minh et al., 2019), all of which score a (fixed) set of trajectories, i.e., trajectory library (Liu & Atkeson, 2009). Moreover, we implement the Deep Imitative Model (Rhinehart et al., 2020, DIM) and an optimistic variant of RIP, termed RIP-BCM and described in Table 1.

### Offline Forecasting Experiments

We use the provided train-val-test splits from (Phan-Minh et al., 2019), for towns Boston and Singapore. For all methods we use $N = 50$ trajectories, and in case of both DIM and RIP, we only optimise the “imitation prior” (cf. Eqn. 4), since goals are not provided, running $N$ planning procedures with different random initializations. The performance of the baselines and our methods are reported on Table 2. We can affirmatively answer Q4 since RIP consistently outperforms the current state-of-the-art methods in out-of-sample evaluation. Moreover, Q2 can be partially answered, since the epistemic-uncertainty-unaware baselines underperformed compared to RIP.

Nonetheless, since we do not have full control over train and test splits at the ICRA 2020 challenge and hence we cannot introduce distribution shifts, we are not able to address questions Q1 and Q3 with the nuScenes benchmark. To that end, we now introduce a control benchmark based on the CARLA driving simulator (Dosovitskiy et al., 2017).
Table 3. We evaluate different autonomous driving methods in terms of their robustness to distribution shifts, in our new benchmark, CARNOVEL. All methods are trained on CARLA Town01 using imitation learning on expert demonstrations from the autopilot (Dosovitskiy et al., 2017). A “‡” indicates methods that use first-person camera view, as in (Chen et al., 2019), a “♣” methods that use LIDAR observation, as in (Ripke et al., 2020) and a “⋆” methods that use the ground truth game engine state, as in (Chen et al., 2019). A “♦” indicates that we used the reference implementation from the original paper, otherwise we used our implementation. For all the scenes we chose pairs of start-destination locations and ran 10 trials with randomised initial simulator state for each pair. Standard errors are in gray (via bootstrap sampling). The outperforming method is in bold. The complete CARNOVEL benchmark results are in Appendix B.

<table>
<thead>
<tr>
<th>Methods</th>
<th>Abnormal Turn (7 x 10 scenes, %)</th>
<th>Success ↑ (4 x 10 scenes, %)</th>
<th>Infra/km ↓ (x1e-3)</th>
<th>Success ↑ (5 x 10 scenes, %)</th>
<th>Infra/km ↓ (x1e-3)</th>
</tr>
</thead>
<tbody>
<tr>
<td>CIL ♦‡ (Codevilla et al., 2018)</td>
<td>65.71 ± 0.37</td>
<td>7.04 ± 0.07</td>
<td>60.00 ± 29.34</td>
<td>4.74 ± 3.02</td>
<td>20.00 ± 0.00</td>
</tr>
<tr>
<td>LbC♦† (Chen et al., 2019)</td>
<td>0.00 ± 0.00</td>
<td>5.81 ± 0.58</td>
<td>50.00 ± 0.00</td>
<td>1.61 ± 0.15</td>
<td>08.00 ± 10.95</td>
</tr>
<tr>
<td>LbC-GT♣ (Chen et al., 2019)</td>
<td>02.86 ± 0.39</td>
<td>3.68 ± 0.34</td>
<td>05.00 ± 11.18</td>
<td>3.36 ± 0.26</td>
<td>00.00 ± 0.00</td>
</tr>
<tr>
<td>DIM ♦ (Rhinehart et al., 2020)</td>
<td>74.28 ± 11.26</td>
<td>5.56 ± 4.06</td>
<td>70.00 ± 10.54</td>
<td>6.87 ± 4.09</td>
<td>20.00 ± 0.42</td>
</tr>
<tr>
<td>RIP-BCM♣ (baseline, cf. Table 1)</td>
<td>68.57 ± 09.03</td>
<td>7.93 ± 3.73</td>
<td>75.00 ± 00.00</td>
<td>5.49 ± 4.03</td>
<td>06.00 ± 09.66</td>
</tr>
<tr>
<td>RIP-MA♣ (ours, cf. Section 3.3.2)</td>
<td>84.28 ± 14.20</td>
<td>7.86 ± 5.70</td>
<td>97.50 ± 07.90</td>
<td>0.26 ± 0.54</td>
<td>38.00 ± 06.32</td>
</tr>
<tr>
<td>RIP-WCM♣ (ours, cf. Section 3.3.1)</td>
<td>87.14 ± 14.20</td>
<td>4.91 ± 3.00</td>
<td>87.50 ± 13.17</td>
<td>1.83 ± 1.73</td>
<td>42.00 ± 06.32</td>
</tr>
</tbody>
</table>

4.2. CARNOVEL

In order to access the robustness of AD methods to novel, OOD driving scenarios, we introduce a benchmark, called CARNOVEL. In particular, CARNOVEL is built on the CARLA simulator (Dosovitskiy et al., 2017). Offline expert demonstrations from Town01 are provided for training. Then, the driving agents are evaluated on a suite of OOD navigation tasks, including but not limited to roundabouts, challenging non-right-angled turns and hills, none of which are experienced during training. The CARNOVEL tasks are summarised in Appendix A. Next, we introduce metrics that quantify and help us answer questions Q1. Q3.

4.2.1. METRICS

Since we are studying navigation tasks, agents should be able to reach safely pre-specified destinations. As done also in previous work (Codevilla et al., 2018; Rhinehart et al., 2020; Shen et al., 2019), the infractions per kilometre metric (i.e., violations of rules of the road and accidents per driven kilometre) measures how safely the agent navigates. The success rate measures the percentage of successful navigations to the destination, without any infraction. However, these standard metrics do not directly reflect the methods’ performance under distribution shifts. As a result, we introduce two new metrics for quantifying the performance in out-of-training distribution tasks:

Detection score. The correlation of infractions and model’s uncertainty termed detection score is used to measure a method’s ability to predict the OOD scenes that lead to catastrophic events. As discussed by Michelmore et al. (2018), we look at time windows of 4 seconds (Taoka, 1989; Coley et al., 2009). A method that can detect potential infractions should have high detection score.

Recovery score. The percentage of successful manoeuvres in novel scenes — where the uncertainty-unaware methods fail — is used to quantify a method’s ability to recover from distribution shifts. We refer to this metric as recovery score. A method that is oblivious to novelty should have 0 recovery score, but positive otherwise.

4.2.2. BASELINES

We compare RIP against the current state-of-the-art imitation learning methods in the CARLA benchmark (Codevilla et al., 2018; Rhinehart et al., 2020; Shen et al., 2019). Apart from DIM and RIP-BCM, discussed in Section 4.1.2, we also benchmark:

Conditional imitation learning (Codevilla et al., 2018, CIL) is a discriminative behavioural cloning method that conditions its predictions on contextual information (e.g., LIDAR) and high-level commands (e.g., turn left, go straight).

Learning by cheating (Shen et al., 2019, LbC) is a method that builds on CIL and uses (cross-modal) distillation of privileged information (e.g., game state, rich, annotated bird-eye-view observations) to a sensorimotor agent. For reference, we also evaluate the agent who has uses privileged information directly (i.e., teacher), which we term LbC-GT.

4.2.3. ONLINE PLANNING EXPERIMENTS

All the methods are trained on offline expert demonstrations from CARLA Town01. We perform 10 trials per CARNOVEL task with randomised initial simulator state and the results are reported on Table 3 and Appendix B.
Our robust imitative planning (i.e., RIP-WCM and RIP-MA) consistently outperforms the current state-of-the-art imitation learning-based methods in novel, OOD driving scenarios. In alignment with the experimental results from nuScenes (cf. Section 4.1), we address questions Q4 and Q2, reaching the conclusion that RIP’s epistemic uncertainty explicit mechanism for recovery improves its performance under distribution shifts, compared to epistemic uncertainty-unaware methods. As a result, RIP’s recovery score (cf. Section 4.2.1) is higher than the baselines.

Towards distribution shift detection and answering questions Q1 and Q3, we collect 50 scenes for each method that led to a crash, record the uncertainty 4 seconds (Taoka, 1989) before the accident and assert if the uncertainties can be used for detection. RIP’s (ours) predictive variance (cf. Eqn. (3)) serves as a useful detector, while DIM’s (Rhinehart et al., 2020) negative log-likelihood was unable to detect catastrophes. The results are illustrated on Figure 5.

Despite RIP’s improvement over current state-of-the-art methods with 97.5% success rate and 0.26 infractions per driven kilometre (cf. Table 3), the safety-critical nature of the task mandates higher performance. Towards this goal, we introduce an online adaptation variant of RIP.

5. Adaptive Robust Imitative Planning

We empirically observe that the quantification of epistemic uncertainty and its use in the RIP objectives is not always sufficient to recover from shifts away from the training distribution (cf. Section 4.2.3). However, we can use uncertainty estimates to ask the human driver to take back control or default to a safe policy, avoiding potential infractions. In the former case, the human driver’s behaviors can be recorded and used to reduce RIP’s epistemic uncertainty via online adaptation. The epistemic uncertainty is reducible and hence it can be eliminated, provided enough demonstrations.

We propose an adaptive variant of RIP, called AdaRIP, which uses the epistemic uncertainty estimates to decide when to query the human driver for feedback, which is used to update its parameters online, adapting to arbitrary new driving scenarios. AdaRIP relies on external, online feedback from an expert demonstrator, similar to DAgger (Ross et al., 2011) and its variants (Zhang & Cho, 2016; Cronrath et al., 2018). However, unlike this prior work, AdaRIP uses an epistemic uncertainty-aware acquisition mechanism. AdaRIP’s pseudocode is given in Algorithm 1.

The uncertainty (i.e., variance) threshold, \( \tau \), is calibrated on a validation dataset, such that it matches a pre-specified level of false negatives, using a similar analysis to Figure 5.

6. Benchmarking Adaptation

The goal of this section is to provide experimental evidence for answering the following questions: Q5. Can RIP’s epistemic-uncertainty estimation be used for efficiently querying an expert for online feedback (i.e., demonstrations)? Q6. Does AdaRIP’s online adaptation mechanism improve success rate?

We evaluate AdaRIP on CARNOVEL tasks, where the CARLA autopilot (Dosovitskiy et al., 2017) is queried for demonstrations online when the predictive variance (cf. Eqn. (3)) exceeds a threshold, chosen according to RIP’s detection score, (cf. Figure 5). We measure performance

---

2AdaRIP is also compatible with other feedback mechanisms, such as expert preferences (Christiano et al., 2017) or explicit reward functions (de Haan et al., 2019).
Can Autonomous Vehicles Identify, Recover From, and Adapt to Distribution Shifts?

According to the:

**Adaptation score.** The improvement in success rate as a function of number of online expert demonstrations is used to measure a method’s ability to adapt efficiently online. We refer to this metric as adaptation score. A method that can adapt online should have a positive adaptation score.

AdaRIP’s performance on the most challenging CARNOVEL tasks is summarised in Figure 4, where, as expected, the success rate improves as the number of online demonstrations increases. Qualitative examples are illustrated in Appendix C.

Although AdaRIP can adapt to any distribution shift, it is prone to catastrophic forgetting and sample-inefficiency, as many online methods (French, 1999). In this paper, we only demonstrate AdaRIP’s efficacy to adapt under distribution shifts and do not address either of these limitations. Future work lies in providing a practical, sample-efficient algorithm to be used in conjunction with the AdaRIP framework. Methods for efficient (e.g., few-shot or zero-shot) and safe adaptation (Finn et al., 2017; Zhou et al., 2019) are orthogonal to AdaRIP and hence any improvement in these fields could be directly used for AdaRIP.

7. Related Work

**Imitation learning.** Learning from expert demonstrations (i.e., imitation learning (Widrow & Smith, 1964; Pomerleau, 1989, IL)) is an attractive framework for sequential decision-making in safety-critical domains such as autonomous driving, where trial and error learning has little to no safety guarantees during training. A plethora of expert driving demonstrations has been used for IL (Caesar et al., 2019; Sun et al., 2019; Kesten et al., 2019) since a model mimicking expert demonstrations can simply learn to stay in “safe”, expert-like parts of the state space and no explicit reward function need be specified.

On the one hand, behavioural cloning approaches (Liang et al., 2018; Sauer et al., 2018; Li et al., 2018; Codevilla et al., 2018; 2019; Chen et al., 2019) fit command-conditioned discriminative sequential models to expert demonstrations, which are used in deployment to produce expert-like trajectories. On the other hand, Rhinehart et al. (2020) proposed command-unconditioned expert trajectory density models which are used for planning trajectories that both satisfy the goal constraints and are likely under the expert model. However, both of these approaches fit point-estimates to their parameters, thus do not quantify their model (epistemic) uncertainty, as explained next. This is especially problematic when estimating what an expert would or would not do in unfamiliar, OOD scenes. In contrast, our methods, RIP and AdaRIP, does quantify epistemic uncertainty in order to both improve planning performance and triage situations in which an expert should intervene.

**Novelty detection & epistemic uncertainty.** A principled means to capture epistemic uncertainty is with Bayesian inference to compute the predictive distribution. However, evaluating the posterior \(p(\theta|D)\) with exact inference is intractable for non-trivial models (Neal, 2012). Approximate inference methods (Graves, 2011; Blundell et al., 2015; Gal & Ghahramani, 2016; Hernández-Lobato & Adams, 2015) have been introduced that can efficiently capture epistemic uncertainty. One approximation for epistemic uncertainty in deep models is model ensembles (Lakshminarayanan et al., 2017; Chua et al., 2018). Prior work by Kahn et al. (2017) and Kenton et al. (2019) use ensembles of deep models to detect and avoid catastrophic actions in navigation tasks, although they can not recover from or adapt to distribution shifts. Our epistemic uncertainty-aware planning objective, RIP, instead, managed to recover from some distribution shifts, as shown experimentally in Section 4.

**Coping with distribution shift.** Strategies to cope with distribution shift include (a) domain randomization; (b) domain adaptation and (c) online adaptation. Domain randomization assumes access to a simulator and exhaustively searches for configurations that cover all the data distribu-
Algorithm 1: Adaptive Robust Imitative Planning

\textbf{Input}: 
\begin{itemize}
  \item $\mathcal{D}$ Demonstrations \quad $I(x_t; s_{t+1})$ Local planner
  \item $K$ Number of models \quad $q(y|x; \theta)$ Imitative model
  \item $\mathcal{B}$ Data buffer \quad $p(G|y)$ Goal likelihood
  \item $\tau$ Variance threshold \quad $p(\theta)$ Model prior
\end{itemize}

// Approximate model posterior inference, e.g., deep ensemble

for model index $k = 1 \ldots K$ do
  Bootstrap sample dataset $D_k \sim \mathcal{D}$
  Sample model parameters from prior, $\theta_k \sim p(\theta)$
  Train ensemble's $k$-component via maximum likelihood estimation (MLE) // Eqn. (1)
  $\theta_k \leftarrow \arg\max_\theta \mathbb{E}_{(x,y) \sim D_k} \left[ \log q(y|x; \theta) \right]$

// Online planning
$x, \mathcal{G} \leftarrow \text{env.reset}()$
while not done do
  Get robust imitative plan // Eqn. (4)
  $y^* \leftarrow \arg\max_\theta \log q(y|x; \theta) + \log p(G|y)$

  // Online adaptation
  Estimate the predictive variance of the $y^*$ plan's quality under the model posterior // Eqn. (3)
  $u(y^*) = \text{Var}_p(\theta|D) \left[ \log q(y^*|x; \theta) \right]$
  if $u(y^*) > \tau$ then
    $y^* \leftarrow \text{Query expert at } x$
    $\mathcal{B} \leftarrow \mathcal{B} \cup (x, y^*)$
    Update model posterior on $\mathcal{B}$ // with any few-shot adaptation method
  $a_t \leftarrow I(y^*)$
  $x, \mathcal{G}, \text{ done} \leftarrow \text{env.step}(a_t)$
end

Hence, our epistemic-uncertainty-aware method, Robust Imitative Planning can cope with some OOD events, thereby reducing the system's dependency on expert feedback, and can use this uncertainty to decide when it cannot cope—when the expert must intervene.

\textbf{Current benchmarks.} We are interested in the control problem, where AD agents get deployed in reactive environments and make sequential decisions. The CARLA Challenge (Ros et al., 2019; Dosovitskiy et al., 2017; Codevilla et al., 2019) is an open-source benchmark for control in AD. It is based on 10 traffic scenarios from the NHTSA pre-crash typology (National Highway Traffic Safety Administration, 2007) to inject challenging driving situations into traffic patterns encountered by AD agents. The methods are only assessed in terms of their generalization to weather conditions, the initial state of the simulation (e.g., the start and goal locations, and the random seed of other agents.) and the traffic density (i.e., empty town, regular traffic and dense traffic).

Despite these challenging scenarios selected in the CARLA Challenge, the agents are allowed to train on the same scenarios in which they evaluated, and so the robustness to distributional shift is not assessed. Consequently, both Chen et al. (2019) and Rhinehart et al. (2020) manage to solve the CARLA Challenge with almost 100% success rate, when trained in Town01 and tested in Town02. However, both methods score almost 0% when evaluated in Roundabouts due to the presence of OOD road morphologies, as discussed in Section 4.2.3.

\section{8. Summary and Conclusions}

To summarise, in this paper, we studied autonomous driving agents in out-of-training distribution tasks (i.e. under distribution shifts). We introduced an epistemic-uncertainty-aware planning method, called robust imitative planning (RIP), which can detect and recover from distribution shifts, as shown experimentally in a real prediction task, nuScenes, and a driving simulator, CARLA. We presented an adaptive variant (AdaRIP) which uses RIP’s epistemic uncertainty estimates to efficiently query the expert for online feedback and adapt its model parameters online. We also introduced and open-sourced an autonomous car novel-scene benchmark, termed CARNOVEL, to assess the robustness of driving agents to a suite of OOD tasks.

\section*{Acknowledgements}

This work was supported by the UK EPSRC CDT in Autonomous Intelligent Machines and Systems (grant reference EP/L015897/1). This project has received funding from the Office of Naval Research, the DARPA Assured Autonomy Program, and ARL DCIST CRA W911NF-17-2-0181, Microsoft Azure and Intel AI Labs.