ECAI 2024
U. Endriss et al. (Eds.)

doi:10.3233/FAIA240843

Informed Spectral Normalized Gaussian Processes for Trajectory Prediction

Christian Schlauch[a,b,*], **Christian Wirth**[a] **and Nadja Klein**[c]

[a]Continental Automotive Technologies GmbH, AI Lab Berlin
[b]Humboldt-Universität zu Berlin
[c]Karlsruhe Institute of Technology, Scientific Computing Center, Methods for Big Data

Abstract. Prior parameter distributions provide an elegant way to represent prior expert knowledge for informed learning. Previous work has shown that using such informative priors to regularize probabilistic deep learning (DL) models increases their performance and data efficiency. However, commonly used sampling-based approximations for probabilistic DL models can be computationally expensive, requiring multiple forward passes and longer training times. Promising alternatives are compute efficient last layer kernel approximations like spectral normalized Gaussian processes (SNGPs). We propose a novel regularization-based continual learning method for SNGPs, which enables the use of informative priors that represent prior knowledge learned from previous tasks. Our proposal builds upon well-established methods and requires no rehearsal memory or parameter expansion. We apply our *informed SNGP* model to the trajectory prediction problem in autonomous driving by integrating prior drivability knowledge. On two public datasets, we investigate its performance under diminishing training data and across locations, and thereby demonstrate an increase in data efficiency and robustness to location-transfers over non-informed and informed baselines.

1 Introduction

Deep learning (DL) has become a powerful artificial intelligence (AI) tool for handling complex tasks. However, DL requires extensive training data to provide robust results [10]. High acquisition costs can render the collection of sufficient data unfeasible. This is especially problematic in safety-critical domains like autonomous driving, where we encounter a wide range of edge cases associated with high risks [30]. Informed learning (IL) aims to improve the data efficiency and robustness of DL models by integrating prior knowledge [28]. Most IL approaches consider prior scientific knowledge, for example the physics of motion, by constraining or verifying the problem space or learning process directly. However, hard constraints are not suitable for qualitative prior expert knowledge since reasonable exceptions can frequently occure. In autonomous driving, for example, we expect traffic participants to comply with speed regulations but must not rule out violations. Still, knowledge about regulations or norms can be highly informative for most cases and are readily available at low cost.

A recent idea is the integration of such prior expert knowledge into probabilistic DL models [25, 23]. These models maintain a distribution over possible model parameters instead of single maximum likelihood estimates. The prior knowledge can be represented as a prior parameter distribution, learned from arbitrarily defined knowledge tasks, to regularize training on real-world observations. The probabilistic informed learning (PIL) approach of Schlauch et al. [23] applies this idea to the trajectory prediction in autonomous driving using regularization-based continual learning methods, achieving a substantially improved data efficiency. However, typical sampling-based probabilistic DL model approximations, such as the variational inference (VI) used by Schlauch et al. [23], are computationally expensive, since they require multiple forward passes and substantially more training epochs. A promising alternative are compute efficient last layer approximations [13]. The spectral normalized Gaussian process (SNGP) [14] is a particularly efficient approximation, that applies a Gaussian process (GP) as last layer to a deterministic deep neural network (DNN). The DNN acts as scalable feature extractor, while the last layer GP allows the deterministic estimation of the uncertainty in a single forward pass. The last layer GP kernel itself is approximated via a finite number of Fourier features, which is easy to scale and asymptotically exact.

We propose a novel regularization-based continual learning method to enable the use of SNGPs in a PIL approach. Our proposal builds upon well-established methods [24, 15], imposes little computational overhead and requires no additional architecture changes, which makes it applicable in a broad range of application domains. We apply our method in a PIL approach for the trajectory prediction in autonomous driving, which is an especially challenging application since well-calibrated, multi-modal predictions are required to enable safe planning.

To be able to compare to existing literature, we follow Schlauch et al. [23] by using CoverNet [19] as base model and by integrating the prior *drivability* knowledge that trajectories are likely to stay on-road. We benchmark our proposed *informed CoverNet-SNGP* on two public datasets, NuScenes and Argoverse2, against the non-informed Base-CoverNet, CoverNet-SNGP and informed Transfer-CoverNet, GVCL-Det-CoverNet as baselines. To this end, we evaluate data efficiency by diminishing the training data availability and robustness to location-transfers. Both data efficiency and robustness are key in developing generalizing prediction models and enabling safe autonomous driving [17, 30]. We observe benefits in favor of our informed CoverNet-SNGP across various performance metrics, especially in low data regimes, which demonstrates our method's viability to increase data efficiency and robustness in a PIL approach.

* Corresponding Author. Email: christian.schlauch@student.hu-berlin.de

In summary, our contributions are:

1. A novel regularization-based continual learning method for compute efficient SNGPs, that enables their use in a PIL approach;
2. an application for trajectory prediction in autonomous driving, integrating prior drivability knowledge into our CoverNet-SNGP;
3. an extensive evaluation on two datasets, showing that we improve on informed and non-informed CoverNet baselines, especially in low data regimes.

Our code is available on GitHub [22].

2 Related Work

von Rueden et al. [28] provides an overview of IL as an emerging field of research, which is also known as knowledge-guided or knowledge-augmented learning [30]. In trajectory prediction, like in other domains, most work concentrates on integrating prior scientific knowledge. Dynamical models are used, for instance, to encode physical limitations of motion in the architecture [7], in the output representation [19] or in a post-hoc verification [2]. Approaches similar to the PIL approach [23], that focus on integrating prior expert knowledge, leverage transfer- or multi-task learning settings [3]. However, transfer learning does not prevent catastrophic forgetting, while multi-task learning requires a single dataset with simultaneously available labels. PIL can be applied without these limitations.

SNGPs and related models, known as deterministic uncertainty models (DUMs), have been analyzed by Postels et al. [20] and Charpentier et al. [5]. Most closely related to SNGPs is the deterministic uncertainty estimator (DUE) proposed by van Amersfoort et al. [27], which approximates the last layer kernel with sparse variational inducing points instead of Fourier features. DUE preserves the non-parametric nature of the kernel, but its point-wise convergence to the true posterior makes it very sensitive to the number and initialization of inducing points.

Parisi et al. [18] and De Lange et al. [8] give a detailed survey of continual learning methods and their classification. Our proposed continual learning method for SNGPs is purely regularization-based, in contrast to the functional regularization introduced by Titsias et al. [26], which could be directly applied to the DUE model, and the work of Derakhshani et al. [9], which also considers a kernel approximation based on Fourier features. Both these methods require rehearsal, the latter also a parameter expansion. Rehearsal is likely to be sensitive to the data imbalances [1] in our application, while parameter expansions require architecture changes which introduce additional complexity. Our proposed method is conceptually simple and builds upon the well-established online elastic weight consolidation (online EWC) introduced by Schwarz et al. [24]. Online EWC can also be understood as special case of generalized variational continual learning (GVCL) described by Loo et al. [15].

3 Informed SNGPs

3.1 Probabilistic Informed Learning

The PIL approach of Schlauch et al. [23] integrates prior expert knowledge in a supervised learning setup. The basic idea is to define a sequence of knowledge tasks $i = 1, \dots, M-1$ on datasets $D_i = \{(x_j^{(i)}, y_j^{(i)})\}_{j=1}^{n_i}$ with n_i samples each. These datasets can be synthetically generated, for example, by leveraging semantic annotations to map the prior knowledge to the prediction target. Semantic annotations are readily available in domains like autonomous driving, but are often underutilized in state-of-the-art models that learn from observations in the conventional task $i = M$ alone [16].

Given a probabilistic DL model parameterized by θ and an initial uninformative prior $\pi_0(\theta)$, the goal is to recursively learn from the sequence of tasks by applying Bayes' rule

$$p(\theta|D_{1:i}) \propto \pi_0(\theta) \prod_{j=1}^{i} p_\theta(y_j|x_j), \tag{1}$$

where $p_\theta(y_j|x_j)$ are the likelihood functions at task j, which are assumed to be conditionally independent given θ. The informative priors make information explicit and shape the loss surface in the downstream task, improving the training outcome [25]. To improve the computational tractability, the recursion is approximated by repurposing regularization-based continual learning methods.

The PIL approach can generally be applied, as long as first, the prior knowledge is strongly related to the observational task, second, the prior knowledge can be mapped to the prediction target and third, the posterior parameter distribution can be estimated [23].

3.2 SNGP Composition

SNGPs [14] employ a composition $f_\theta = g_{\theta_{\text{GP}}} \circ h_{\theta_{\text{NN}}} : \mathcal{X} \to \mathcal{Y}$, $\theta = \{\theta_{\text{NN}}, \theta_{\text{GP}}\}$. Its first component is a deterministic, spectral normalized feature extractor $h_{\theta_{\text{NN}}} : \mathcal{X} \to \mathcal{H}$ with trainable parameters θ_{NN} mapping the high dimensional input space $\mathcal{X}$ into a low dimensional hidden space $\mathcal{H}$. The spectral normalization ensures a distance-sensivity of the mapping by approximatively constraining the Lipschitz constant of the residual blocks between 0 and some upper bound s. The second component is a GP output layer $g_{\theta_{\text{GP}}} : \mathcal{H} \to \mathcal{Y}$ with a radial basis function (RBF) kernel mapping into the output space $\mathcal{Y}$. The RBF kernel can be approximated by a finite number of (random) Fourier features using Bochner's Theorem [21]. This effectively reduces the GP to a Bayesian linear model, that can be written as a neural network layer with fixed hidden weights and trainable output weight parameters θ_{GP} and enables end-to-end training with the feature extractor. The distance-sensitivity of the composition, due to the spectral normalization and RBF kernel choice, prevents a "feature-collapse" [27], which improves the calibration against adversarial and outlier samples.

In total, SNGP introduces five additional hyperparameters, namely an upper bound s and number of power iterations N_p for the spectral normalization for the feature extractor and the number of Fourier features N_{GP}, the kernel's length scale l_s and Gaussian prior choice for the output weights θ_{GP}.

3.3 Regularizing SNGPs

There are two problems prohibiting the direct application of the PIL approach to composite last layer kernel approximations like the SNGP. First, there is no existing continual learning method for kernels that does not require rehearsal memories or parameter expansions (see Sec. 2). Second, estimating the posterior parameter distribution of the feature extractor (e.g. via a Laplace approximation or variational inference) contradicts the motivation for the last layer kernel approximation regarding compute-efficiency.

We tackle the first problem by leveraging the Fourier feature approximation of the RBF kernel of the GP. The posterior distributions of the parameters of the last layer at task i can be made tractable

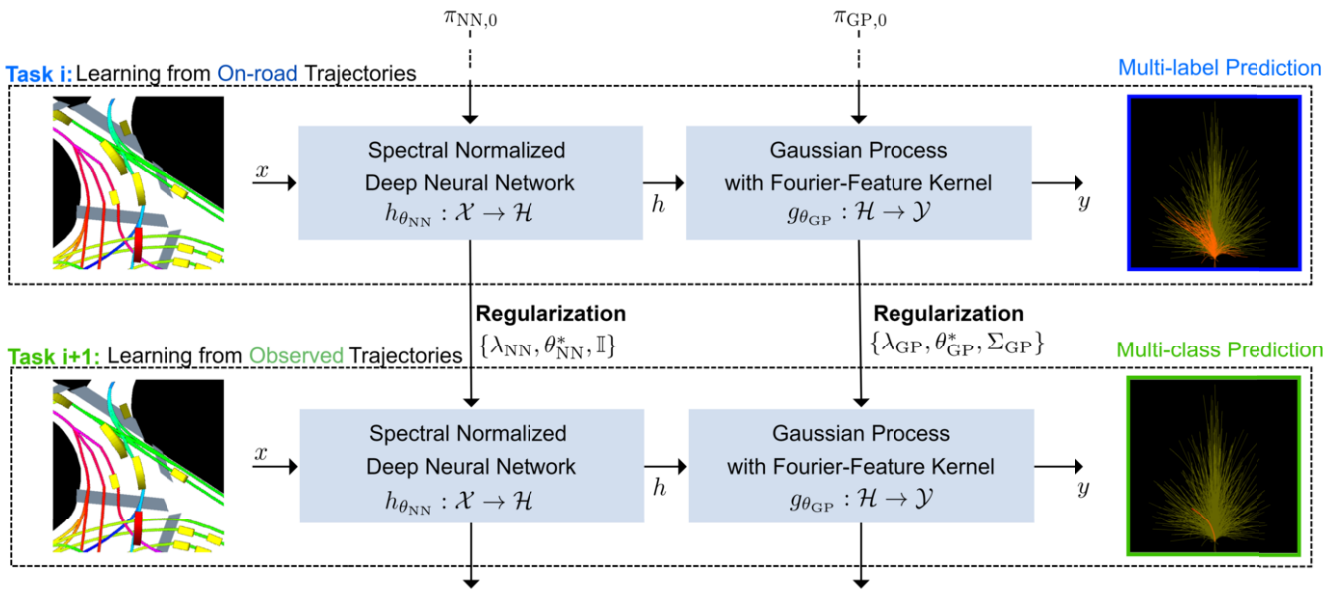

Figure 1: The informed CoverNet-SNGP model consists of a spectral normalized feature extractor and a last layer Gaussian Process with a Fourier feature approximated radial basis function kernel. Given a Birds-Eye-View RGB rendering and the target's current state, the model classifies a set of candidate trajectories according to their drivability in task i and their likely realization in task $i+1$. Our method regularizes the training on task $i+1$, given the MAP estimates and Laplace approximated covariance from task i as informative priors, thereby integrating the drivability knowledge following the PIL approach.

through Laplace approximation [14], that is, we assume

$$p(\theta_{\text{GP}}|D_{1:i}) \approx \mathcal{N}(\theta_{\text{GP}}; \theta^*_{\text{GP},i}, \Sigma^{-1}_{\text{GP},i}),$$

given a maximum a posteriori (MAP) estimate $\theta^*_{\text{GP},i}$ at task i. Similar to online EWC [24], $\theta^*_{\text{GP},i}$ can be obtained by minimizing

$$-\log p_{\theta_{\text{GP}}}(y_i|x_i) - \frac{\lambda_{\text{GP}}}{2}(\theta_{\text{GP}} - \theta^*_{\text{GP},i-1})^\top \Sigma^{-1}_{\text{GP},i-1}(\theta_{\text{GP}} - \theta^*_{\text{GP},i-1}) \quad (2)$$

with respect to θ_{GP}, where the precision $\Sigma^{-1}_{\text{GP},i}$ is approximated by the sum of the Hessian at the MAP estimate and a scaled precision at task $i-1$, that is,

$$\Sigma^{-1}_{\text{GP},i} \approx H_{\text{GP},i}(\theta^*_{\text{GP},i}) + \gamma_{\text{GP}}\Sigma^{-1}_{\text{GP},i-1}. \quad (3)$$

Above, $\lambda_{\text{GP}} > 0$ is a temperature parameter, that scales the importance of the previous task [12], and $0 < \gamma_{\text{GP}} \leq 1$ is a decay parameter, that allows for more plasticity over very long task sequences [24]. In contrast to online EWC, we can cheaply compute the Hessian using moving averages [14] instead of using a Fisher matrix approximation. To tackle the second problem and regularize the feature extractor, we approximate the precision $\Sigma^{-1}_{\text{NN},i-1}$ with the identity matrix $\mathbb{I}$. This implies a $\mathcal{L}2$-regularization for the MAP estimates $\theta^*_{\text{NN},i}$ obtained by minimizing

$$-\log p_{\theta_{\text{NN}}}(y_i|x_i) - \frac{\lambda_{\text{NN}}}{2}(\theta_{\text{NN}} - \theta^*_{\text{NN},i-1})^2 \quad (4)$$

with respect to θ_{NN}, where λ_{NN} is the extractor specific temperature parameter. This idea is conceptually simple, but should be sufficient, since the learned representation in knowledge tasks should be suitable downstream due to the assumed close relation between tasks.

In the first task $i = 1$, we can use an uninformative zero-mean, unit-variance prior $\pi_{\text{GP},0}$ for the GP layer and $\pi_{\text{NN},0}$ for the feature extractor, which amounts to a simple $\mathcal{L}2$-regularization in both cases. This is a common choice in the probabilistic deep learning literature, implying that the trainable parameters are a priori independent and equally important [12, 15].

In result, the complete model $f_\theta : \mathcal{X} \to \mathcal{Y}$, parameterized by $\theta = \{\theta_{\text{NN}}, \theta_{\text{GP}}\}$, can be effectively regularized and used in the PIL approach, as visualized in Figure 1. Our method introduces three hyperparameters $\{\lambda_{\text{GP}}, \gamma_{\text{GP}}, \lambda_{\text{NN}}\}$. It only requires the parameters of the previous task in memory and has little computational overhead like online EWC [24].

4 Application to Trajectory Prediction

4.1 Problem Definition

We limit ourselves to the *single-agent* trajectory prediction problem [11]. An autonomous driving system is assumed to observe the states in the state space $\mathcal{Y}$ of all agents $\mathcal{A}$ present in a scene on the road. Let $y^{(t)} \in \mathcal{Y}$ denote the state of target agent $a \in \mathcal{A}$ at time t and let $y^{(t-T_o:t)} = \left(y^{(t-T_o)}, y^{(t-T_o+\delta t)}, \ldots, y^{(t)}\right)$ be its observed trajectory over an observation history T_o with sampling period δt. Additionally, we assume access to agent-centered maps $\mathcal{M}$, which include semantic annotations such as the drivable area. Map and states make up the scene context of agent a, denoted as $x = (\{y_j^{(t-T_o:t)}\}_{j=1}^{|\mathcal{A}|}, \mathcal{M})$. Given x, the goal is to predict the distribution of a's future trajectories $p(y^{(t+\delta t:t+T_h)}|x)$ over the prediction horizon T_h, where $y^{(t-\delta t:t+T_h)} = \left(y^{(t+\delta t)}, y^{(t+2\delta t)}, \ldots, y^{(t+T_h)}\right)$.

4.2 CoverNet-SNGP

CoverNet [19] approaches the single-agent trajectory problem by considering a birds-eye-view RGB rendering of the scene context x and the current state $y^{(t)}$ of the target agent a as inputs. The RGB rendering is processed by a computer-vision backbone, before being concatenated with the target's current state and processed by another

dense layer. The output is represented as a set $\mathcal{K}$ of K candidate trajectories $y_k^{(t+\delta t:t+T_h)}$. Doing so reduces the prediction problem to a classification problem, where each trajectory in the set $\mathcal{K}$ is treated as a sample of the predictive distribution $p(y^{(t+\delta t:t+T_h)}|x)$ and only the conditional probability of each sample is required. In principle, any heuristic that leads to an exhaustive set of possible trajectories may be used to define $\mathcal{K}$. For example, we may use a dynamical model that also integrates physical limitations [19], which could be applied in combination with the PIL approach. For simplicity, we follow Phan-Minh et al. [19] definition of a fixed set $\mathcal{K}$ by solving a set-covering problem over a subsample of observed trajectories in the training split, using a greedy-algorithm[1] given a coverage-bound ϵ, which determines the number of total candidates K.

The modification of CoverNet with SNGP is straightforward if a convolutional neural network (CNN) is used as backbone. In that case, a spectral normalization can be directly applied to the architecture while the last layer is replaced with a Gaussian process, approximated by Fourier features as described in Sec. 3.2.

4.3 Integrating Prior Drivability Knowledge

The PIL approach is applied sequentially on two consecutive tasks as follows. In task i, we integrate the prior drivability knowledge, that trajectories are likely to stay on-road. To this end, we derive new training labels (see Sec. 3.1), where all candidate trajectories in $\mathcal{K}$ with way-points inside the drivable area for a given training scene x are labeled as positive [3]. We then train in a multi-label classification with a binary cross-entropy loss on these labels. In task $i+1$, the closest candidate trajectory in $\mathcal{K}$ to the observed ground truth is labeled as positive. We then train in a multi-class classification with a sparse categorical cross-entropy loss (using softmax normalized logit transformations) on these labels [19]. In effect, the consecutive tasks are only differing in the labels and loss functions used. Applying our method described in Sec. 3.3, we first train our CoverNet-SNGP model on task i and then regularize its training on task $i+1$, as exemplified in Figure 1. We denote the resulting informed CoverNet-SNGP as CoverNet-SNGP$_{\mathbf{I}}$, opposed to the non-informed version CoverNet-SNGP$_{\mathbf{U}}$ trained on task $i+1$ only, without integration of prior knowledge from task i.

5 Experimental Design

5.1 Datasets

We use the public NuScenes [4] and Argoverse2 [29] datasets. We replicate the NuScenes data split by Phan-Minh et al. [19] on Argoverse2, only considering vehicle targets (exlcuding pedestrians and cyclists not driving on-road), as summarized in Table 1. For the RGB rendering, we consider each scene with a one-second history ($T_o = 1$s). For the candidate trajectories in $\mathcal{K}$, we consider a six-second prediction horizon ($T_h = 6$s), sampled at 2Hz in NuScenes and 10Hz in Argoverse2. Both datasets include drivable areas in the semantic map data, enabling the first task as described in Sec. 4.3.

5.2 Baselines

We consider the unmodified CoverNet as baseline, once as non-informed Base-CoverNet [19] and once as informed Transfer-CoverNet. The Transfer-CoverNet baseline, pretrained on task i and then trained on the current task $i+1$, has previously been proposed by Boulton et al. [3]. We can also understand it as an ablation-type baseline to the PIL approach without regularization. In addition, we compare to GVCL-Det-CoverNet proposed by Schlauch et al. [23], since it only needs a single forward pass at test time, too.

[1] Further details in our supplemental on Github [22]. Also see Chapter 35.3 of Cormen et al. [6] regarding set-covering problems in general.

Table 1: Numbers and percentages of samples across location subsets of both NuScenes and Argoverse2.

data subset	train split # (%)	train-val split # (%)	val split # (%)
NuScenes Total	32186 (100.0)	8560 (100.0)	9041 (100.0)
Boston	19629 (60.99)	5855 (68.40)	5138 (56.84)
Singapore	12557 (49.01)	2705 (31.60)	3903 (43.16)
Argoverse2 Total	161379 (100.0)	22992 (100.0)	23113 (100.0)
Miami	42214 (26.16)	5983 (26.02)	5984 (25.89)
Austin	34681 (21.49)	4968 (21.57)	4985 (26.16)
Pittsburgh	33391 (20.69)	4823 (20.98)	4803 (20.78)
Dearborn	20579 (12.75)	2933 (12.79)	3001 (12.98)
Washington-DC	20546 (12.73)	2883 (12.54)	2976 (12.88)
Palo-Alto	9968 (6.18)	1402 (6.10)	1364 (5.90)

5.3 Metrics

We measure the average displacement error minADE_1 and final displacement error minFDE_1, evaluating the quality of the most likely trajectory, and the minADE_5, which considers the five most likely trajectories [11]. The minADE_5 depends on the probability-based ordering and, thus, indirectly on the calibration. We also consider the drivable area compliance (DAC) to evaluate the extent to which predictions align with our prior drivability knowledge.

Since observed ground truth trajectories may not be part of the trajectory set $y_{\text{true}}^{(t+\delta t:t+T_h)} \notin \mathcal{K}$, the CoverNet model exhibits an irreducible approximation error. To more clearly assess the impact of our method, we also consider the classification-based negative log likelihood (NLL) and the rank of the positively labeled trajectory (RNK), both directly depending on the calibration, and the Top1-accuracy (ACC).

5.4 Implementation Details

We use the output representation described in Sec. 4 with a coverage bound $\epsilon = 4$m, for NuScenes with $K_{\text{Nusc}} = 415$ and for Argoverse2 with $K_{\text{Argo}} = 518$ candidates. We employ a ResNet-50 as backbone and SGD as optimizer. For the CoverNet-SNGPs, we fix power iterations N_p to one and the number of Fourier features N_f to 1024, following Liu et al. [14]. The spectral normalization's upper bound s and the kernel length scale l_s are treated as additional hyperparameters. We tune the hyperparameters of each model on the respective tasks with 100% of the data using the validation NLL[2]. The exception is CoverNet-SNGP$_{\mathbf{I}}$, which uses the same settings as CoverNet-SNGP$_{\mathbf{U}}$ on task $i+1$. We also fix both temperature parameters λ_{NN} and λ_{GP} ad-hoc to the inverse of the effective dataset size to keep tuning costs low. The decay parameter γ_{GP} is mostly relevant for very long task sequences (see Sec. 3), such that we set $\gamma_{\text{GP}} = 1$.

6 Results

We study the performance of our CoverNet-SNGP$_{\text{I}}$ against the baselines under two sets of experiments. First, we investigate the performance under increasingly smaller subsets of the observational training data, allowing us to shed light on data efficiency. These subsets are randomly subsampled once and then kept fixed across models and

[2] Configurations are available in our supplemental on Github [22].

Table 2: Average performance and standard deviation of 5 independent repetitions over decreasing subsamples of NuScenes (**bold** as best).

Data (in %)	Model	minADE_1	minADE_5	minFDE_1	NLL	RNK	ACC (in %)	DAC (in %)
100	Base	4.92 ±0.15	2.34 ±0.05	10.94 ±0.27	3.47 ±0.06	15.55 ±0.73	13.94 ±1.10	89.26 ±1.13
	Transfer	4.60 ±0.04	**2.18 ±0.02**	9.94 ±0.08	3.21 ±0.01	**11.79 ±0.18**	15.19 ±0.43	**95.73 ±0.29**
	GVCL-Det*	4.55 ±0.11	2.26 ±0.05	**9.93 ±0.39**	3.60 ±0.08	11.85 ±0.48	14.88 ±0.94	90.94 ±2.25
	SNGP_U	4.53 ±0.09	2.25 ±0.04	10.31 ±0.27	3.23 ±0.01	13.25 ±0.19	17.04 ±0.68	91.19 ±0.61
	SNGP_I	**4.45 ±0.04**	2.21 ±0.01	10.09 ±0.12	**3.19 ±0.01**	12.44 ±0.14	**17.36 ±0.59**	91.65 ±0.59
50	Base	5.15 ±0.23	2.37 ±0.11	11.46 ±0.60	3.52 ±0.06	17.21 ±1.33	13.55 ±0.62	86.68 ±4.72
	Transfer	4.86 ±0.04	2.26 ±0.01	10.38 ±0.06	3.35 ±0.01	13.46 ±0.21	14.37 ±0.09	**95.66 ±0.28**
	SNGP_U	4.57 ±0.05	2.26 ±0.04	10.40 ±0.15	3.30 ±0.02	14.62 ±0.17	**16.83 ±0.59**	90.09 ±0.56
	SNGP_I	**4.48 ±0.07**	**2.22 ±0.04**	**10.13 ±0.13**	**3.25 ±0.02**	**13.39±0.31**	16.72 ±0.76	91.10 ±0.72
30	Base	5.40 ±0.03	2.44 ±0.07	12.01 ±0.20	3.68 ±0.04	19.80 ±1.03	12.70 ±0.81	86.58 ±2.54
	Transfer	5.08 ±0.03	2.34 ±0.02	10.80 ±0.07	3.47 ±0.01	15.00 ±0.05	13.38 ±0.31	**96.07 ±0.32**
	SNGP_U	4.68 ±0.09	**2.29 ±0.04**	10.61 ±0.26	3.37 ±0.01	16.02 ±0.22	16.69 ±0.62	89.22 ±0.30
	SNGP_I	**4.58 ±0.03**	2.30 ±0.02	**10.35 ±0.08**	**3.31 ±0.02**	**14.67 ±0.20**	**17.10 ±0.34**	90.41 ±0.49
10	Base	5.89 ±0.28	2.72 ±0.11	12.88 ±0.63	3.99 ±0.06	32.74 ±1.48	12.38 ±0.96	86.38 ±2.64
	Transfer	6.09 ±0.03	2.65 ±0.02	12.60 ±0.06	3.89 ±0.01	24.82 ±0.13	10.35 ±0.15	**95.54 ±0.23**
	GVCL-Det*	5.27 ±0.27	2.53 ±0.09	12.03 ±0.58	4.05 ±0.07	24.78 ±0.45	12.95 ±0.80	91.52 ±1.54
	SNGP_U	5.00 ±0.04	2.52 ±0.03	11.36 ±0.22	3.60 ±0.02	25.19 ±0.30	**15.73 ±0.22**	88.62 ±0.56
	SNGP_I	**4.96 ±0.05**	**2.47 ±0.04**	**11.25 ±0.16**	**3.52 ±0.03**	**20.94 ±0.59**	15.39 ±0.29	89.53 ±1.11
5	Base	5.90 ±0.17	2.82 ±0.06	12.81 ±0.38	4.26 ±0.03	42.55 ±1.92	10.17 ±1.26	86.89 ±1.96
	Transfer	6.62 ±0.04	2.89 ±0.01	13.41 ±0.09	4.30 ±0.01	29.74 ±0.44	8.70 ±0.14	**97.46 ±0.07**
	SNGP_U	5.07 ±0.05	2.58 ±0.02	11.63 ±0.13	3.90 ±0.03	31.77 ±0.85	14.29 ±0.30	86.31 ±0.82
	SNGP_I	**5.01 ±0.04**	**2.53 ±0.04**	**11.43 ±0.10**	**3.72 ±0.03**	**25.99 ±0.65**	**14.32 ±0.59**	86.85 ±0.94
3	Base	6.23 ±0.16	3.11 ±0.11	13.32 ±0.28	4.53 ±0.03	59.34 ±3.76	10.42 ±0.71	84.83 ±2.00
	Transfer	7.52 ±0.09	3.35 ±0.07	14.71 ±0.14	4.61 ±0.01	36.62 ±0.60	7.33 ±0.10	**97.80 ±0.08**
	GVCL-Det*	6.12 ±0.11	2.86 ±0.09	13.25 ±0.31	4.26 ±0.05	31.96 ±3.01	10.87 ±0.49	93.05 ±1.21
	SNGP_U	5.56 ±0.09	2.85 ±0.07	12.64 ±0.15	4.61 ±0.02	46.82 ±1.37	**13.37 ±0.09**	86.00 ±0.95
	SNGP_I	**5.44 ±0.13**	**2.74 ±0.06**	**12.38 ±0.29**	**3.90 ±0.01**	**27.88 ±1.54**	12.62 ±0.64	86.18 ±0.77
1	Base	8.39 ±1.16	3.44 ±0.30	16.25 ±2.27	5.23 ±0.10	83.20 ±3.84	5.39 ±1.92	81.48 ±4.34
	Transfer	8.44 ±0.07	4.18 ±0.08	15.71 ±0.27	5.52 ±0.01	52.92 ±0.11	4.53 ±0.15	**98.29 ±0.27**
	SNGP_U	6.33 ±0.64	2.88 ±0.07	12.76 ±1.07	5.48 ±0.01	77.64 ±3.38	8.48 ±0.92	70.42 ±3.97
	SNGP_I	**5.39 ±0.28**	**2.68 ±0.07**	**12.27 ±0.53**	**4.40 ±0.01**	**50.19 ±1.15**	**9.94 ±0.56**	79.34 ±2.54

Table 3: Average performance and standard deviation of 5 independent repetitions over decreasing subsamples of Argoverse2 (**bold** as best).

Data (in %)	Model	minADE_1	minADE_5	minFDE_1	NLL	RNK	ACC (in %)	DAC (in %)
100	Base	3.57 ±0.07	1.84 ±0.04	8.96 ±0.15	2.73±0.01	7.87 ±0.58	24.46 ±0.59	94.77 ±0.32
	Transfer	3.60 ±0.04	**1.76 ±0.02**	8.78 ±0.08	**2.68 ±0.01**	**7.31 ±0.18**	24.61 ±0.33	**96.91 ±0.09**
	SNGP_U	3.60 ±0.04	1.86 ±0.03	9.00 ±0.15	2.74 ±0.03	8.19 ±0.28	25.24 ±0.29	95.00 ±0.47
	SNGP_I	**3.51 ±0.06**	1.82 ±0.03	**8.73 ±0.07**	2.69 ±0.01	7.69 ±0.17	**25.56 ±0.42**	95.01 ±0.12
50	Base	3.93 ±0.10	1.97 ±0.07	9.78 ±0.31	2.98 ±0.07	9.89 ±0.53	21.02 ±1.17	93.99 ±1.58
	Transfer	3.80 ±0.01	**1.83 ±0.02**	**9.36 ±0.05**	**2.80 ±0.01**	**8.16 ±0.02**	23.41 ±0.29	**97.04 ±0.22**
	SNGP_U	3.84 ±0.04	2.01 ±0.02	9.67 ±0.15	2.89 ±0.02	9.95 ±0.28	23.53 ±0.40	94.93 ±0.19
	SNGP_I	**3.76 ±0.02**	1.95 ±0.02	9.38 ±0.06	2.84 ±0.02	9.27 ±0.15	**23.81 ±0.76**	94.92 ±0.64
30	Base	4.22 ±0.10	2.04 ±0.01	10.41 ±0.28	3.07 ±0.06	11.18 ±1.33	19.76 ±0.49	94.37 ±0.71
	Transfer	3.99 ±0.02	**1.89 ±0.02**	**9.76 ±0.05**	**2.91 ±0.01**	**9.07 ±0.04**	22.02 ±0.23	**97.15 ±0.23**
	SNGP_U	3.98 ±0.03	2.09 ±0.04	9.95 ±0.09	2.99 ±0.01	11.40 ±0.30	**22.79 ±0.22**	94.73 ±0.56
	SNGP_I	**3.96 ±0.04**	2.04 ±0.02	9.88 ±0.12	2.95 ±0.02	10.54 ±0.18	22.60 ±0.30	94.94 ±0.58
10	Base	4.70 ±0.10	2.25 ±0.02	11.43 ±0.16	3.42 ±0.06	17.17 ±0.28	16.91 ±0.38	93.63 ±0.52
	Transfer	4.49 ±0.02	**2.07 ±0.02**	10.76 ±0.05	3.21 ±0.01	**12.40 ±0.07**	18.46 ±0.27	**97.48 ±0.55**
	SNGP_U	4.26 ±0.03	2.23 ±0.03	10.49 ±0.22	3.19 ±0.02	14.99 ±0.13	20.19 ±0.29	94.22 ±0.89
	SNGP_I	**4.23 ±0.08**	2.22 ±0.05	**10.47 ±0.20**	**3.15 ±0.02**	13.85 ±0.38	**20.90 ±0.16**	94.35 ±0.65
5	Base	5.04 ±0.09	2.41 ±0.06	12.33 ±0.23	3.67 ±0.02	23.73 ±0.87	15.05 ±0.80	90.79 ±1.79
	Transfer	4.94 ±0.01	2.25 ±0.01	11.49 ±0.02	3.50 ±0.01	16.80 ±0.03	15.86 ±0.16	**97.12 ±0.38**
	SNGP_U	4.43 ±0.04	2.31 ±0.02	11.06 ±0.08	3.36 ±0.02	18.92 ±0.29	19.32 ±0.29	91.60 ±0.82
	SNGP_I	**4.41 ±0.01**	**2.24 ±0.02**	**10.92 ±0.11**	**3.28 ±0.01**	**16.30 ±0.30**	**19.36 ±0.48**	93.17 ±1.20
3	Base	5.41 ±0.09	2.48 ±0.11	12.99 ±0.47	3.88 ±0.03	28.85 ±1.35	13.55 ±0.47	90.96 ±0.50
	Transfer	5.44 ±0.01	2.44 ±0.07	12.35 ±0.04	3.73 ±0.01	20.89 ±0.04	13.81 ±0.07	**97.16 ±0.33**
	SNGP_U	4.54 ±0.04	2.34 ±0.02	11.31 ±0.13	3.50 ±0.01	21.96 ±0.34	17.78 ±0.22	91.28 ±0.82
	SNGP_I	**4.51 ±0.05**	**2.33 ±0.04**	**11.06 ±0.14**	**3.41 ±0.01**	**18.04 ±0.33**	**17.94 ±0.20**	92.49 ±0.79
1	Base	5.96 ±0.26	2.75 ±0.04	14.15 ±0.62	4.46 ±0.01	50.60 ±0.99	11.33 ±0.96	87.43 ±3.49
	Transfer	6.52 ±0.03	2.95 ±0.01	14.30 ±0.06	4.28 ±0.01	33.31 ±0.12	10.02 ±0.02	**98.70 ±0.03**
	SNGP_U	5.02 ±0.05	2.53 ±0.02	12.26 ±0.13	3.96 ±0.01	40.58 ±0.50	**15.14 ±0.50**	89.11 ±0.83
	SNGP_I	**5.00 ±0.09**	**2.50 ±0.02**	**12.14 ±0.19**	**3.75 ±0.02**	**26.63 ±0.90**	15.12 ±0.39	90.34 ±0.64

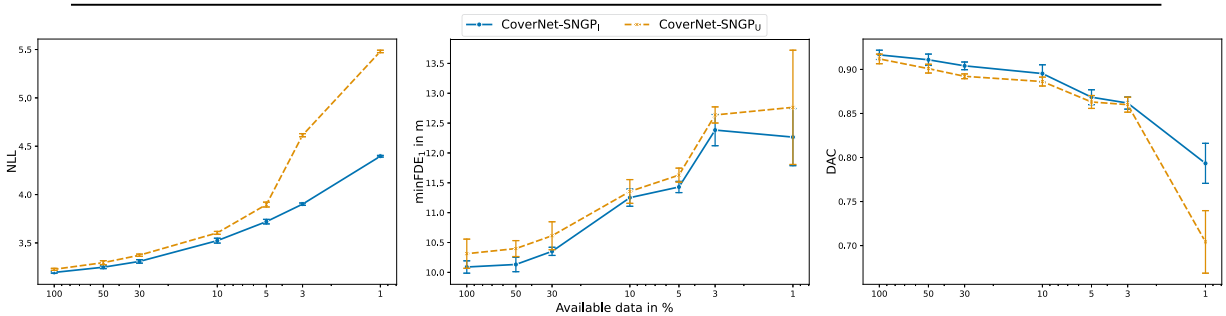

Figure 2: Average performance and standard deviation in NLL, minFDE_1 and DAC of five repetitions for the informed and non-informed CoverNet-SNGP over decreasing subsamples of NuScenes.

Table 4: Average performance and standard deviation of 5 independent repetitions trained on Singapore and Boston locations from NuScenes.

Train Location	Model	Test Location	minADE_1	minADE_5	minFDE_1	NLL	RNK	ACC	DAC
Singapore	Base	Singapore	5.33 ±0.40	2.37 ±0.06	11.54±0.80	3.69±0.06	19.79±0.71	12.97±1.89	84.94±1.35
		Boston	5.83 ±0.22	**2.64 ±0.04**	12.76±0.41	3.93±0.07	24.98±1.08	10.18±0.80	89.79±2.13
	Transfer	Singapore	5.47 ±0.07	2.35 ±0.03	11.41±0.14	3.49±0.02	13.79±0.18	11.49±0.50	**94.29±0.59**
		Boston	6.65 ±0.10	2.94 ±0.03	14.26 ±0.20	4.09 ±0.01	24.09 ±0.31	8.55 ±0.40	**96.09±0.20**
	SNGP_U	Singapore	4.48 ±0.06	2.26 ±0.02	10.05 ±0.16	3.38 ±0.03	15.06 ±0.31	**15.85 ±0.46**	85.30 ±1.01
		Boston	5.38 ±0.15	2.71 ±0.05	12.20 ±0.35	**3.65 ±0.02**	**20.81 ±0.56**	**13.15±0.73**	90.37±0.95
	SNGP_I	Singapore	**4.43 ±0.07**	**2.20 ±0.06**	**9.83 ±0.15**	**3.31 ±0.06**	**13.64 ±1.31**	15.84±1.05	86.56±0.50
		Boston	**5.36 ±0.09**	2.68 ±0.08	**12.18 ±0.26**	**3.65 ±0.01**	21.56±1.40	12.95 ±0.74	90.68 ±0.79
Boston	Base	Boston	5.02 ±0.20	2.32 ±0.09	11.18±0.49	3.57 ±0.08	18.10±1.31	13.18±1.23	90.18 ±2.13
		Singapore	5.69 ±0.28	2.73 ±0.15	12.77 ±0.78	3.88 ±0.07	23.42 ±0.47	11.37 ±0.98	82.03 ±2.92
	Transfer	Boston	4.78 ±0.06	2.19 ±0.01	10.21 ±0.12	3.41 ±0.01	14.39 ±0.19	14.02 ±0.56	**96.50 ±0.74**
		Singapore	5.63 ±0.06	2.64 ±0.04	12.17 ±0.20	3.70 ±0.01	18.77 ±0.16	11.40 ±0.61	**93.10 ±1.15**
	SNGP_U	Boston	4.62 ±0.10	2.23 ±0.02	10.46 ±0.25	3.32 ±0.01	14.83 ±0.40	16.57 ±0.64	93.31 ±0.40
		Singapore	4.94 ±0.07	2.61 ±0.11	11.27 ±0.17	3.58 ±0.03	19.48 ±0.16	14.93 ±1.06	83.28 ±0.60
	SNGP_I	Boston	**4.50 ±0.04**	**2.19 ±0.02**	**10.13 ±0.11**	**3.26 ±0.02**	**12.97 ±0.33**	**16.94 ±0.60**	94.01 ±0.27
		Singapore	**4.82 ±0.07**	**2.60 ±0.07**	**10.95 ±0.18**	**3.52 ±0.04**	**18.39 ±0.66**	**15.60 ±0.70**	85.36 ±1.18

Table 5: Average performance and standard deviation of 5 independent repetitions trained on Palo-Alto and Miami locations from Argoverse2.

Train Location	Model	Test Location	minADE_1	minADE_5	minFDE_1	NLL	RNK	ACC	DAC
Palo-Alto	Base	Palo-Alto	4.94 ±0.12	2.35 ±0.05	12.13 ±0.20	3.45 ±0.07	17.41 ±0.21	14.72 ±1.01	92.94 ±1.41
		Ex-Palo-Alto	5.02 ±0.42	2.51 ±0.23	12.24 ±0.51	3.65 ±0.12	22.18 ±1.20	14.18 ±0.79	91.90 ±1.53
	Transfer	Palo-Alto	4.91 ±0.05	**2.19 ±0.01**	11.32 ±0.13	3.27 ±0.01	13.75 ±0.13	18.66 ±0.43	**95.92 ±0.38**
		Ex-Palo-Alto	5.33 ±0.03	2.44 ±0.01	12.39 ±0.90	3.63 ±0.01	18.30 ±0.13	13.68 ±0.34	**95.92 ±0.46**
	SNGP_U	Palo-Alto	**4.23 ±0.06**	2.20 ±0.01	10.63 ±0.19	3.11 ±0.03	15.03 ±0.56	**23.42 ±0.35**	92.02 ±1.74
		Ex-Palo-Alto	**4.55 ±0.05**	2.38 ±0.02	11.35 ±0.13	3.37 ±0.02	18.66 ±0.55	**18.58 ±0.68**	92.06 ±1.55
	SNGP_I	Palo-Alto	**4.23 ±0.05**	**2.19 ±0.04**	**10.40 ±0.22**	**3.06 ±0.03**	**13.72 ±0.54**	22.38 ±0.47	91.74 ±3.01
		Ex-Palo-Alto	4.57 ±0.11	**2.37 ±0.04**	**11.30 ±0.32**	**3.35 ±0.01**	**17.43 ±0.54**	18.09 ±0.74	91.88 ±2.25
Miami	Base	Miami	4.02 ±0.21	2.22 ±0.11	10.28 ±0.35	3.45 ±0.06	14.12 ±0.78	18.97 ±0.31	95.20 ±0.98
		Ex-Miami	4.29 ±0.22	2.31 ±0.13	11.01 ±0.39	3.47 ±0.09	16.18 ±0.92	17.92 ±0.79	94.99 ±1.12
	Transfer	Miami	3.91 ±0.01	**1.85 ±0.01**	**9.52 ±0.02**	**2.94 ±0.01**	**9.17 ±0.04**	21.33 ±0.29	**97.42 ±0.72**
		Ex-Miami	4.31 ±0.02	**2.07 ±0.01**	10.47 ±0.05	3.10 ±0.01	**10.62 ±0.04**	19.65 ±0.35	**97.41 ±0.98**
	SNGP_U	Miami	**3.88 ±0.04**	2.03 ±0.02	9.74 ±0.11	3.00 ±0.01	11.48 ±0.16	**22.07 ±0.41**	95.58 ±0.40
		Ex-Miami	**4.15 ±0.04**	2.21 ±0.02	10.44 ±0.13	3.11 ±0.01	13.56 ±0.21	**21.50 ±0.51**	94.81 ±0.35
	SNGP_I	Miami	**3.88 ±0.05**	1.99 ±0.02	9.65 ±0.15	2.99 ±0.01	10.75 ±0.21	21.71 ±0.53	95.21 ±0.46
		Ex-Miami	4.17 ±0.05	2.20 ±0.03	**10.42 ±0.15**	**3.09 ±0.02**	12.68 ±0.31	21.25 ±0.59	94.26 ±0.58

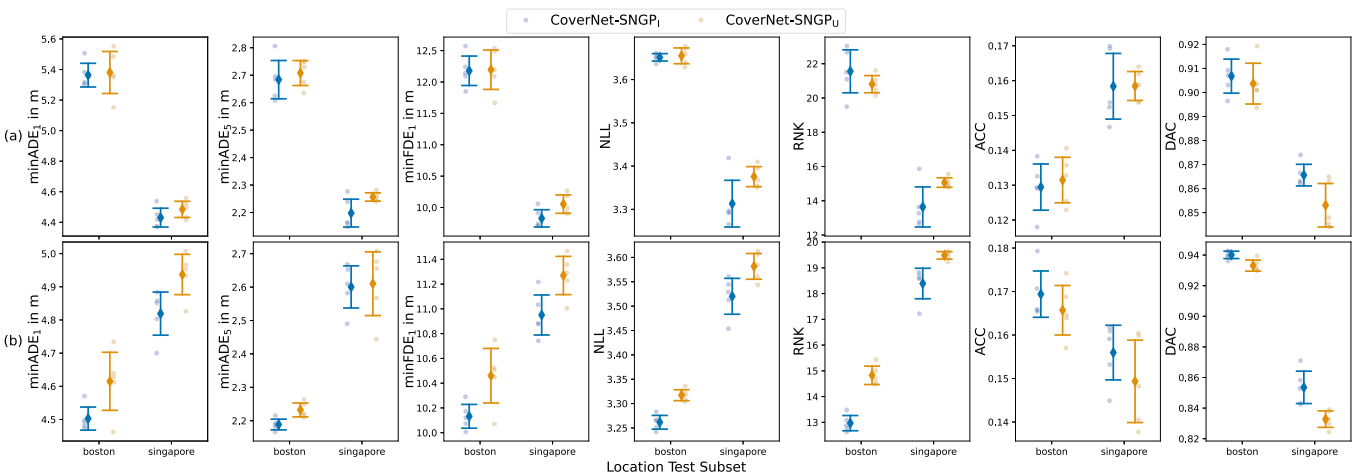

Figure 3: Average performance and standard deviation of the informed and non-informed CoverNet-SNGP on Boston and Singapore test data, with (a) models trained on Singapore training data and (b) models trained on Boston training data (five repetitions).

repetitions. In this set, we also consider GVCL-Det-CoverNet with results on NuScenes for 100%, reported from Schlauch et al. [23], 10% and 3%, replicated with only three independent repetitions, due to the long training times. Second, we test the performance by training and testing on location-specific subsets, gaining insights into the robustness to location-transfers, which is often implicitly assumed in the state of the art [17]. The reported results are the average performance and standard deviation of five independent runs for each experiment.

6.1 *Effect of Available Training Data*

Table 2 and Table 3 show the performance of our CoverNet-SNGP_I in comparison to the baselines on NuScenes and Argoverse2, respectively. Across baselines, the performance seems relatively stable when only half the data is available, but diminishes increasingly with less available training data. We argue, that this is a symptom of the dataset composition. The datasets contain a higher proportion of repetitive scenes (e.g. driving on straight roads) and performance deteriorations in the long tail of edge cases are not immediately visible in metric averages over the whole test set until low data regimes are reached.

Notably, the prior drivability knowledge leads to performance benefits in our CoverNet-SNGP_I and informed baselines (Transfer-CoverNet, GVCl-Det-CoverNet) across most metrics. The benefits from the prior drivability knowledge are most substantial in the calibration-sensitive metrics (RNK and notably NLL, e.g., as seen in Figure 2) that directly benefit from the optimization in the knowledge tasks. The drivability knowledge is less helpful in discerning the best candidate between the remaining drivable candidate trajectories, leading to lower benefits in the respective metrics (minADE_1, minFDE_1, ACC).

We also observe, that Transfer-CoverNet's benefits are limited to

higher data regimes. In low data regimes, Transfer-CoverNet can even perform substantially worse than Base-CoverNet across all metrics (except DAC). In these low data regimes, Transfer-CoverNet may converge to less adequate minima, due to its weight initialization being overly biased towards drivability (illustrated by the rising DAC). In contrast, GVCL-Det-CoverNet and our CoverNet-SNGP$_\text{I}$ never decrease performance, with consistent benefits especially in low data regimes. This highlights a principal advantage of the PIL approach, where the informative prior helps to shape the complete loss landscape during training.

In comparison to GVCL-Det-CoverNet, our CoverNet-SNGP$_\text{I}$ shows benefits across most metrics, especially in low data regimes, even though both are trained using the PIL approach. The advantage is most visible in the metrics concerning the most-likely trajectory (minADE$_1$, ACC). CoverNet-SNGP$_\text{I}$ also shows more stable results with lower standard deviations. Here, our CoverNet-SNGP$_\text{I}$ profits from using the full information of the posterior distribution at test time.

6.2 *Effect of Location-Specific Training*

Table 4 and Table 5 show location-specific performances of our CoverNet-SNGP$_\text{I}$ in comparison to the baselines on NuScenes and Argoverse2, respectively. We observe, that the performance generally and substantially deteriorates in locations which are not included in the training data. This sensitivity of trajectory prediction models to location-transfers can be a major limitation to their practical use.

We also observe, that our CoverNet-SNGP$_\text{I}$ can help to alleviate this issue by consistently improving the generalization over location-transfers. This is most visible in the comparison of the Boston trained models on NuScenes (see Figure 3) and the Palo-Alto trained models in Argoverse2, where we see a better performance across most metrics in same-location and location-transfer tests. The Transfer-CoverNet baseline performs even worse than Base-CoverNet in these cases, pointing to the same limitation we see in Sec. 6.1 regarding its bias. In the other two comparisons, CoverNet-SNGP$_\text{I}$ still shows advantages (notably NLL). However, in case of Miami in Argoverse2, more training data is available (compare Sec. 6.1), and in case of Singapore in NuScenes the drivability knowledge might be less useful (see Figure 3), since all models achieve a lower DAC.

6.3 *Runtime Considerations*

To underline the motivation for compute efficient SNGP approximation, we stress the runtime differences to the sampling-based VI approximation used by Schlauch et al. [23] in Table 6. A key advantage is the reduced overhead at test time. The GVCL-Det-CoverNet baseline also requires computationally extremely expensive training of a GVCL-CoverNet model. For example, in our setting, training until convergence with 10% of NuScenes data needs around 120 hours for GVCL-CoverNet, in contrast to 8 hours for CoverNet-SNGP$_\text{I}$ and 6 hours for Base-CoverNet, due to the lower computational efficiency and higher number of necessary training epochs.

Table 6: Parameter counts and runtimes on single Nvidia RTX A5000 GPU for complete training/evaluation with 100% NuScenes.

Model	**Training** s/epoch	**Latency** ms/sample	**Learnable Parameter** # Million
Base-CoverNet	467	5.6	33
CoverNet-SNGP$_\text{U}$	599	7.0	32
CoverNet-SNGP$_\text{I}$	602	7.0	32
GVCL-CoverNet	851	67.2	67

7 Conclusion

Our work introduces a novel regularization-based continual learning method for the SNGP model. We apply this method in a PIL approach for trajectory prediction in autonomous driving, deriving a compute efficient informed CoverNet-SNGP model integrating prior drivability knowledge. We demonstrate on two public datasets, that our informed CoverNet-SNGP increases data efficiency and robustness to location-transfers, outperforming informed and non-informed baselines in low data regimes. Thus, we show that our proposed continual learning method is a feasible way to regularize SNGPs using informative priors.

Our results leave many perspectives for future research: Since we make minimal assumptions, our method could be utilized in other applications domain. Employing it in a PIL approach is especially promising in domains such as medical research, where rich prior expert knowledge is available and data is scarce. We are also interested in applying our informed SNGPs to more recent transformer-based architectures using self-supervised learning, to investigate the interaction with strong representation learning. Lastly, we aim to evaluate the influence of our method on the robustness to adversarial attacks and outliers.

Ethics Statement

The presented approach allows to bias deep learning models. In the trajectory prediction for autonomous driving, it is arguably beneficial to integrate present biases in societal regulations to better reflect the behavior on the street. Practitioners, however, should be wary of accidentally integrating socially adverse biases.

Acknowledgements

The research leading to these results is funded by the German Federal Ministry for Economic Affairs and Climate Action within the project “KI Wissen – Entwicklung von Methoden für die Einbindung von Wissen in maschinelles Lernen”. The authors would like to thank the consortium for the successful cooperation.

References

[1] B. Bagus and A. Gepperth. An investigation of replay-based approaches for continual learning. In *IEEE International Joint Conference on Neural Networks, IJCNN 2021*.

[2] M. Bahari, I. Nejjar, and A. Alahi. Injecting Knowledge in Data-driven Vehicle Trajectory Predictors. *Transportation Research Part C: Emerging Technologies*, 2021.

[3] F. A. Boulton, E. C. Grigore, and E. M. Wolff. Motion Prediction using Trajectory Sets and Self-Driving Domain Knowledge. *arXiv preprint, https://arxiv.org/abs/2006.04767*, 2021.

[4] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom. nuscenes: A multimodal dataset for autonomous driving. In *Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020*.

[5] B. Charpentier, C. Zhang, and S. Günnemann. Training, architecture, and prior for deterministic uncertainty methods. *arXiv preprint, https://arxiv.org/abs/2303.05796*, 2023.

[6] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. *Introduction to Algorithms, 3rd Edition*. MIT Press, 2009. ISBN 978-0-262-03384-8.

[7] H. Cui, T. Nguyen, F. Chou, T. Lin, J. Schneider, D. Bradley, and N. Djuric. Deep kinematic models for kinematically feasible vehicle trajectory predictions. In *Proceedings of the 2020 IEEE International Conference on Robotics and Automation, ICRA 2020*, 2020.

[8] M. De Lange, R. Aljundi, M. Masana, S. Parisot, X. Jia, A. Leonardis, G. G. Slabaugh, and T. Tuytelaars. A continual learning survey: Defying forgetting in classification tasks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.

[9] M. M. Derakhshani, X. Zhen, L. Shao, and C. Snoek. Kernel continual learning. In *Proceedings of the 38th International Conference on Machine Learning, ICML 2021*.

[10] T. Freiesleben and T. Grote. Beyond generalization: a theory of robustness in machine learning. *Springer Synthese*, 2023.

[11] Y. Huang, J. Du, Z. Yang, Z. Zhou, L. Zhang, and H. Chen. A Survey on Trajectory-Prediction Methods for Autonomous Driving. *IEEE Transactions on Intelligent Vehicles*, 2022.

[12] J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska, D. Hassabis, C. Clopath, D. Kumaran, and R. Hadsell. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences of the United States of America*, 2017.

[13] A. Kristiadi, M. Hein, and P. Hennig. Being bayesian, even just a bit, fixes overconfidence in relu networks. In *Proceedings of the 37th International Conference on Machine Learning, , ICML 2020*, 2020.

[14] J. Z. Liu, Z. Lin, S. Padhy, D. Tran, T. Bedrax-Weiss, and B. Lakshminarayanan. Simple and principled uncertainty estimation with deterministic deep learning via distance awareness. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020*, 2020.

[15] N. Loo, S. Swaroop, and R. E. Turner. Generalized variational continual learning. In *9th International Conference on Learning Representations, ICLR 2021*. OpenReview.net.

[16] O. Makansi, J. von Kügelgen, F. Locatello, P. V. Gehler, D. Janzing, T. Brox, and B. Schölkopf. You mostly walk alone: Analyzing feature attribution in trajectory prediction. In *10th International Conference on Learning Representations, ICLR 2022*. OpenReview.net.

[17] A. Malinin, N. Band, Y. Gal, M. J. F. Gales, A. Ganshin, G. Chesnokov, A. Noskov, A. Ploskonosov, L. Prokhorenkova, I. Provilkov, V. Raina, V. Raina, D. Roginskiy, M. Shmatova, P. Tigas, and B. Yangel. Shifts: A dataset of real distributional shift across multiple large-scale tasks. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks 1, NeurIPS Datasets and Benchmarks 2021*.

[18] G. I. Parisi, R. Kemker, J. L. Part, C. Kanan, and S. Wermter. Continual lifelong learning with neural networks: A review. *Elsevier Neural Networks*, 2019.

[19] T. Phan-Minh, E. C. Grigore, F. A. Boulton, O. Beijbom, and E. M. Wolff. Covernet: Multimodal behavior prediction using trajectory sets. In *Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020*, 2020.

[20] J. Postels, M. Segù, T. Sun, L. D. Sieber, L. V. Gool, F. Yu, and F. Tombari. On the practicality of deterministic epistemic uncertainty. In *Proceedings of the 39th International Conference on Machine Learning, ICML 2022*, Proceedings of Machine Learning Research. PMLR, 2022.

[21] A. Rahimi and B. Recht. Random features for large-scale kernel machines. In *Advances in Neural Information Processing Systems 20: Annual Conference on Neural Information Processing Systems 2007, NeurIPS 2007*.

[22] C. Schlauch and C. Wirth. Code and supplemental. URL https://github.com/continental/kiwissen-bayesian-trajectory-prediction.

[23] C. Schlauch, C. Wirth, and N. Klein. Informed priors for knowledge integration in trajectory prediction. In *Machine Learning and Knowledge Discovery in Databases: Research Track - European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases, ECML PKDD 2023, Turin, Italy*. Springer, 2023.

[24] J. Schwarz, W. Czarnecki, J. Luketina, A. Grabska-Barwinska, Y. W. Teh, R. Pascanu, and R. Hadsell. Progress & compress: A scalable framework for continual learning. In J. G. Dy and A. Krause, editors, *Proceedings of the 35th International Conference on Machine Learning, ICML 2018*.

[25] R. Shwartz-Ziv, M. Goldblum, H. Souri, S. Kapoor, C. Zhu, Y. LeCun, and A. G. Wilson. Pre-train your loss: Easy bayesian transfer learning with informative priors. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022*.

[26] M. K. Titsias, J. Schwarz, A. G. de G. Matthews, R. Pascanu, and Y. W. Teh. Functional regularisation for continual learning with gaussian processes. In *8th International Conference on Learning Representations, ICLR 2020*. OpenReview.net.

[27] J. R. van Amersfoort, L. Smith, A. Jesson, O. Key, and Y. Gal. On feature collapse and deep kernel learning for single forward pass uncertainty. 2021.

[28] L. von Rueden, S. Mayer, K. Beckh, B. Georgiev, S. Giesselbach, R. Heese, B. Kirsch, J. Pfrommer, A. Pick, R. Ramamurthy, M. Walczak, J. Garcke, C. Bauckhage, and J. Schuecker. Informed Machine Learning – A Taxonomy and Survey of Integrating Knowledge into Learning Systems. *IEEE Transactions on Knowledge and Data Engineering*, 2021.

[29] B. Wilson, W. Qi, T. Agarwal, J. Lambert, J. Singh, S. Khandelwal, B. Pan, R. Kumar, A. Hartnett, J. K. Pontes, D. Ramanan, P. Carr, and J. Hays. Argoverse 2: Next generation datasets for self-driving perception and forecasting. *arXiv preprint, https://arxiv.org/abs/2301.00493*, 2023.

[30] J. Wörmann, D. Bogdoll, E. Bührle, H. Chen, E. F. Chuo, K. Cvejoski, L. van Elst, T. Gleißner, P. Gottschall, S. Griesche, C. Hellert, C. Hesels, S. Houben, T. Joseph, N. Keil, J. Kelsch, H. Königshof, E. Kraft, L. Kreuser, K. Krone, T. Latka, D. Mattern, S. Matthes, M. Munir, M. Nekolla, A. Paschke, M. A. Pintz, T. Qiu, F. Qureishi, S. T. R. Rizvi, J. Reichardt, L. von Rueden, S. Rudolph, A. Sagel, G. Schunk, H. Shen, H. Stapelbroek, V. Stehr, G. Srinivas, A. T. Tran, A. Vivekanandan, Y. Wang, F. Wasserrab, T. Werner, C. Wirth, and S. Zwicklbauer. Knowledge Augmented Machine Learning with Applications in Autonomous Driving: A Survey. *arXiv preprint, https://arxiv.org/abs/2205.04712*, 2022.