Probabilistic Symmetry for Multi-Agent Dynamics

Sophia Sun   SOPHIASUN@ENG.UCSD.COM
University of California, San Diego

Robin Walters   R.WALTERS@NORTHEASTERN.EDU
Northeastern University

Jinxi Li   JINXI.LI@CONNECT.POLYU.HK
Hong Kong Polytechnic University

Rose Yu   ROSEYU@UCSD.EDU
University of California, San Diego

Editors: N. Matni, M. Morari, G. J. Pappas

Abstract

Learning multi-agent dynamics is a core AI problem with broad applications in robotics and autonomous driving. While most existing works focus on deterministic prediction, producing probabilistic forecasts to quantify uncertainty is critical for downstream decision-making tasks such as motion planning and collision avoidance. By leveraging the internal symmetry in multi-agent dynamics, specifically rotational equivariance, we can improve not only the accuracy, but also calibration of our probabilistic forecasts. We propose a novel deep dynamics model, Probabilistic Equivariant Continuous COnvolution (PECCO) for probabilistic prediction of multi-agent trajectories. PECCO extends equivariant continuous convolution to model the joint velocity distribution of multiple agents. It uses dynamics integration to propagate the uncertainty from velocity to position. We introduce Energy Score, a proper scoring rule, to evaluate probabilistic predictions. On both synthetic and real-world datasets, PECCO shows significant improvements in accuracy and calibration compared to non-equivariant baselines.

Our code is released at https://github.com/Rose-STL-Lab/PECCO. The appendix of the paper can be accessed at https://arxiv.org/abs/2205.01927.

Keywords: Multi-Agent Modeling, Probabilistic forecasting, deep dynamics model, uncertainty quantification, equivariant neural networks

1. Introduction

Predicting the future trajectory of multiple agents is a critical task with applications in autonomous driving (Chang et al., 2019), social behavioral modeling (Sun et al., 2021). In practice, the problem is difficult due to the inherent stochasticity of human motion, and the strong inter-dependency among the agents where the number of interactions grows quadratically with the number of agents. Moreover, agent movements are often influenced by environmental features such as road markings, boundaries, and social preference, which are impossible to measure and model effectively. Such a partially observed setting introduces a significant amount of uncertainty.

Many recent works on learning multi-agent dynamics has shifted to probabilistic modeling as a principled framework to represent uncertainty (Tang and Salakhutdinov, 2019; Salzmann et al., 2020). However, common metrics used in probabilistic trajectory predictions works, such as minimum

average displacement of 6 samples (minADE) , do not fully reflect the quality of probabilistic forecasts. A probabilistic prediction should be *calibrated* and *sharp*; that is, the predicted distribution must cover likely future scenarios without being so broad and uncertain as to be useless.

In this paper, we propose a **P**robabilistic **E**quivariant **C**ontinuous **CO**nvolutional model (PECCO). PECCO is an equivariant probabilistic trajectory prediction model. Our key insight is to exploit *symmetry* to estimate multidimensional conditional distributions with limited data. We assume the predicted probability distribution is rotation and translation equivariant. That is, if the input data is transformed, the probability distribution will also be likewise transformed. In Figure 1, we see the same car approaching an intersection from either the south or east. The scenes are related by a $\pi/2$ rotation. As the absolute compass direction is not particularly meaningful for local trajectory prediction, the model should thus output the same probability distributions over future trajectories for the car coming from the east as that coming from the north, but rotated by $\pi/2$. Rotational equivariance not only allows our model to produce physically consistent predictions, the multiplicative nature of equivariance also allows us to model a probability space with a smaller sample size (Bloem-Reddy and Teh, 2020). For each sample which an equivariant model is trained on, an equivariant model learns as if it were trained on all transformations of that sample by the symmetry group (Wang et al., 2021).

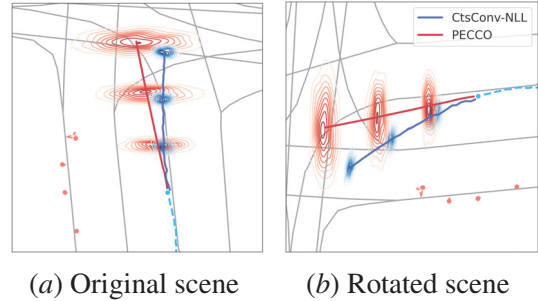(*a*) Original scene   (*b*) Rotated scene

Figure 1: Prediction on the same scene rotated by 90 degrees. PECCO is consistent in trajectory and uncertainty prediction, whereas the non-equivariant model (CtsConv) fails.

PECCO also mitigates issues posed by other methods for enforcing equivariance such as data augmentation and normalization. Data augmentation adds rotated versions of data samples to the training dataset such that the model learns rotational equivariance. However, this slows training drastically, requires greater model capacity, and rarely achieves the level of equivariance or accuracy as equivariant neural networks (Salzmann et al., 2020). Data normalization is a technique that rotates the scene to the agent's reference frame for each data sample, as in Gao et al. (2020). However, in the multi-agent setting, it is impossible to rotate the scene for multiple agents without a canonical reference frame. PECCO allows the weights to be relative to the local orientation of each agent without the need to rotate the scene repeatedly.

Our main contributions are two folds: (1) We design an equivariant neural network, PECCO, for probabilistic forecast of multi-agent dynamics, and (2) we demonstrate that by incorporating symmetry, PECCO improves both calibration and sharpness of probabilistic forecasts on a synthetic particle dataset and two real-world benchmark datasets.

2. Related Work

Trajectory Prediction. Multi-agent trajectory forecasting has been extensively studied, approaches ranged from Kalman filters (Kalman, 1960) to non-linear Gaussian Process Regression models (Jordan, 1998). However, these methods either rely on strong assumptions of the dynamics, or do not explicitly model multi-agent interactions. We refer readers to Rudenko et al. (2020) for

a comprehensive survey of such methods. Advancements in deep learning have allowed flexible modeling of trajectory dynamics (Alahi et al., 2016; Wang et al., 2018; Deo and Trivedi, 2018; Sadeghian et al., 2019; Liang et al., 2020; Walters et al., 2021; Gao et al., 2020; Roddenberry et al., 2021), but they focus mainly on point estimation without uncertainty.

Recent methods have shifted to predicting distributions of future trajectories, capturing uncertainty in dynamics. There are two main categories for probabilistic forecasting: (1) *explicitly* via exact likelihood (Tang and Salakhutdinov, 2019; Chai et al., 2019; Gu et al., 2021) and variational inference (Sohn et al., 2015; Salzmann et al., 2020; Lee et al., 2017), or (2) *implicitly* with Generative Adversarial Networks (GANs) (Gupta et al., 2018; Liu et al., 2019). Our work falls into the first category where we model the distributions parametrically. Parametric models allow us to evaluate the likelihood of future trajectories, which are useful for downstream planning tasks (Chai et al., 2019; Schwarting et al., 2018).

Despite the development in probabilistic modeling, there is no standard metric for quantifying uncertainty of the prediction. Negative log likelihood often overfits the distribution (Guo et al., 2017), and best-of-n-sample results do not evaluate the full distribution (Ivanovic and Pavone, 2021). We argue that probabilistic forecasts should accurately reflect the uncertainty in the model predictions. We propose using proper scoring rules such as Energy Score or mean interval score (Gneiting and Raftery, 2007) for evaluating probabilistic forecasts.

Equivariant Deep Learning. Geometric deep learning that leverages invariance and symmetries has found wide applications in areas ranging from image recognition (Bao and Song, 2019; Worrall et al., 2017; Weiler and Cesa, 2019) to reinforcement learning (van der Pol et al., 2020). Equivariant neural networks are studied for modeling dynamics as well - Fuchs et al. (2020) use $SE(3)$-equivariant transformers to predict trajectories for a small number of particles as a regression task, and Walters et al. (2021) proposed a $S0(2)$ equivariant continuous convolution for traffic trajectory prediction. All the methods mentioned above are deterministic. Köhler et al. (2020) and Satorras et al. (2021) studies equivariant normalizing flows for modeling symmetric densities, however their domains focus on generative modeling and therefore differ from our work significantly. To our knowledge, no previous work has studied equivariant neural networks for probabilistic dynamics forecasting.

Uncertainty Quantification (UQ). Uncertainty quantification is critical for risk assessment in safety-critical domains. Properly quantified uncertainties can be used to create probabilistic constraints and generate more robust planning and control strategies (Ostafew et al., 2016; Bujarbaruah et al., 2019). With the increasing use of deep learning in forecasting tasks, many works have UQ for neural networks (Luo et al., 2021; Wu et al., 2021; Guo et al., 2017). Stankevičiūtė et al. (2021) proposes a conformal prediction algorithm for 1D RNN forecasters with a prediction region with coverage guarantees. However, these works focus only on classification or 1-dimensional forecasts. Salinas et al. (2019, 2020) use autoregressive RNNs for probabilistic forecasting of multiple time series, however, their method cannot explicitly model spatial relations. We present a model design for multi-agent dynamics and produce probabilistic distributions with better calibration.

3. Background

We give a short background on symmetry and equivariance and their probabilistic extension.

Symmetry and Equivariance. A *symmetry group* G is a set together with a composition operation $\circ\colon G \times G \to G$ which is associative and has an identity and inverses. The group G can transform a

vector space V by specifying a *representation* which is a mapping $\rho\colon G \to \mathrm{GL}_n(V)$ sending each element of the group G to an invertible $n \times n$ matrix such that $\rho(g_1 \circ g_2) = \rho(g_1)\rho(g_2)$.

Given a function $f\colon X \to Y$ such that G has representations ρ_X and ρ_Y acting on X and Y respectively, we say f is *G-equivariant* if for all $x \in X$ and $g \in G$, we have $\rho_Y(g)f(x) = f(\rho_X(g)x)$. That is, a transformation of the input of x induces a corresponding transformation of the output. *Invariance* for the function f is a special case in which $\rho_Y(g)y = y$.

$\mathrm{SO}(2)$ Equivariant Continuous Convolution. Continuous convolution (Ummenhofer et al., 2019) generalizes discrete convolution. The feature vector $\mathbf{f}^{(i)} \in \mathbb{R}^{c_{\mathrm{in}}}$ of particle i forms a vector field $\mathbf{f}$, and the kernel of the convolution $K : \mathbb{R}^2 \mapsto \mathbb{R}^{c_{\mathrm{out}} \times c_{\mathrm{in}}}$ forms a matrix field: for each point $\mathbf{x} \in \mathbb{R}^2$, $K(\mathbf{x})$ is a $c_{\mathrm{out}} \times c_{\mathrm{in}}$ matrix. The continuous convolution is then defined by

$$\mathbf{g}^{(i)} = \sum_j a(\|\mathbf{x}^{(j)} - \mathbf{x}^{(i)}\|)K(\mathbf{x}^{(j)} - \mathbf{x}^{(i)}) \cdot \mathbf{f}^{(j)}.$$

By Weiler and Cesa (2019), this is $\mathrm{SO}(2)$-equivariant if $K(gv) = \rho_{\mathrm{out}}(g)K(v)\rho_{\mathrm{in}}(g^{-1})$.

ECCO (Walters et al., 2021) defines the convolution kernel K in polar coordinates $K(\theta, r)$. Let $\mathbb{R}^{c_{\mathrm{in}}}$ and $\mathbb{R}^{c_{\mathrm{out}}}$ be $\mathrm{SO}(2)$-representations ρ_{in} and ρ_{out} respectively, then the convolution kernel satisfies the equivariance condition as follows, making the continuous convolution $\mathrm{SO}(2)$-equivariant.

$$K(\theta + \phi, r) = \rho_{\mathrm{out}}(\mathrm{Rot}_\theta)K(\phi, r)\rho_{\mathrm{in}}(\mathrm{Rot}_\theta^{-1})$$

Calibration and Sharpness of Probabilistic Prediction. It is desirable for a probabilistic prediction to be both *calibrated* and *sharp*. A model is *calibrated* when the predicted probability correspond to the true probability of an event. In forecasting, calibration is often measured with *coverage*, the probability of ground truth Y falls into prediction region of confidence $\hat{C}^{1-\alpha}$, $1 - \alpha \in [0, 1]$.

$$P(Y \in \hat{C}^{1-\alpha}) \geq 1 - \alpha \tag{1}$$

Note that one can always obtain a calibrated prediction region by setting $\hat{C}^{1-\alpha}$ to be the entire Y space. We introduce the concept of *Sharpness*, which refers to the size of the prediction region $|\hat{C}^{1-\alpha}|$. The balance between calibration and sharpness can be measured by a class of metrics called proper scoring rules (Gneiting and Raftery, 2007).

4. PECCO

4.1. Problem Setup

Given past trajectories of n agents over t time steps $\{x_j^{(1)}, x_j^{(2)}, \cdots, x_j^{(n)}\}_{j=1}^t$, where $x_j^{(i)} \in \mathbb{R}^2$, and the environmental context information $\mathbf{e}$ including marker positions of map lane boundaries, we model the probability distribution of agents' positions over k future time steps as $p_\theta(x_{t+1:t+k}|x_{1:t}, \mathbf{e})$, with $x_j = (x_j^{(1)}, \cdots, x_j^{(n)})$ being the positions of all agents at time step j. We introduce PECCO, a deep learning model that leverages rotational equivariance to produce probabilistic forecasts.

The high-level architecture of our model is illustrated in Figure 2. PECCO takes as input the positions of all agents $x_{1:t}$ in the past, and the covariance matrix $\Sigma_{x,j}$ at time j. It outputs the probability distribution of each agents' velocity as a 2-D Gaussian $\mathcal{N}(\mu_{v,j+1}^{(i)}, \Sigma_{v,j+1}^{(i)})$ for the next time step. The velocity distribution is then integrated into a position distribution $\mathcal{N}(\mu_{x,j+1}^{(i)}, \Sigma_{x,j+1}^{(i)})$ through dynamics integration. PECCO predicts the future k timesteps autoregressively. The output distributions are guaranteed to be rotational equivariant by our model implementation.
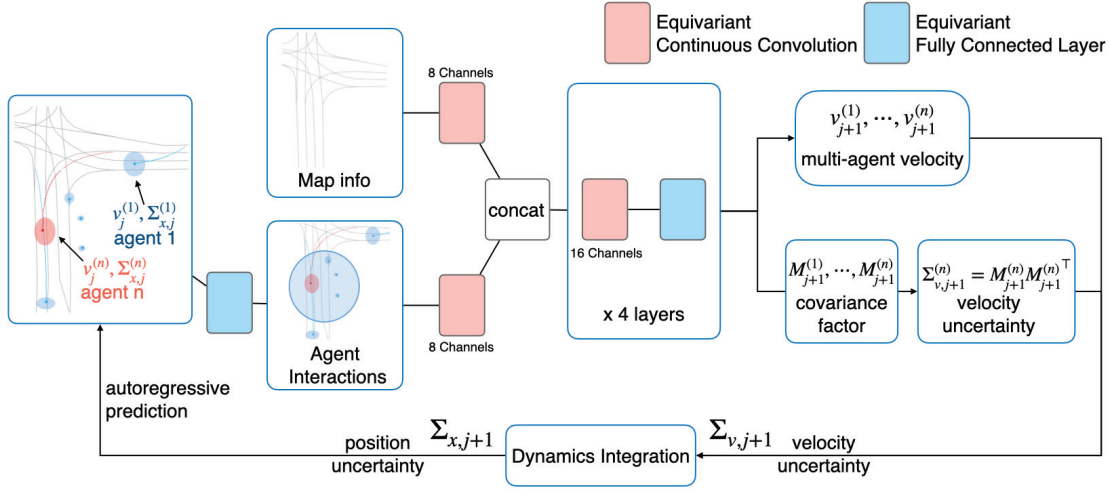
Figure 2: Overview of PECCO's model architecture. Agent trajectories consisting of velocities and position uncertainties are encoded along with map information by equivariant continuous convolution and fully connected layers. The model outputs v_{j+1} and $\Sigma_{v,j+1}$ for all agents, which we use to calculate position uncertainty $\Sigma_{x,j+1}$ via dynamics. The model takes in the forecast and predicts autoregressively.

4.2. Probabilistic Symmetry through Equivariant Neural Networks

Rotational equivariance effectively reduces the dimension of the data space by placing different samples in the same equivalence class. This improves data coverage for better probabilistic modeling.

Intuitively, real-world trajectory dynamics has intrinsic symmetry. That is, if past trajectories and environmental data such as the map is rotated, then the probability of a rotated future trajectory will be equally likely. We can model the probability density function p_θ as an invariant function of its inputs as in Equation 2. Here, each past and future position $x_j^{(i)} \in \mathbb{R}^2$ is transformed according the standard representation ρ_1.

$$p_\theta(x_{t+1:t+k}|x_{1:t}, \mathbf{e}) = p_\theta(gx_{t+1:t+k}|gx_{1:t}, g\mathbf{e}) \quad \forall g \in SO(2) \tag{2}$$

In order to implement Equation 2, we assume future positions follow a multivariate normal distribution $x_j^{(i)} \sim N(\mu_{x,j}^{(i)}, \Sigma_{x,j}^{(i)})$. This is a common assumption in trajectory forecasting literature (Rudenko et al., 2020) and provides a convenient parametric form for optimization and reasoning. In the following expositions we omit the underscore x for simplicity.

We aim to construct an equivariant neural network f_θ that outputs the parameters $\mu_j^{(i)}$ and $\Sigma_j^{(i)}$ autoregressively, taking as input probability distributions over the positions of all agents in the past k time steps

$$\mu_{j+1}, \Sigma_{j+1} = f_\theta(\mu_{j-t+1:j}, \Sigma_{j-t+1:j}, \mathbf{e}).$$

where $\mu_j = (\mu_j^{(1)}, \ldots, \mu_j^{(n)})$ and $\Sigma_j = (\Sigma_j^{(1)}, \ldots, \Sigma_j^{(n)})$ and $\mathbf{e}$ denotes environmental information.

In this case, the equivariance of f_θ leads to the desired invariance of p_θ. This may be seen as a partial evaluation or currying of the conditional probability density function which has the effect of transforming invariance to equivariance. The following proposition relates equivariant networks with probabilistic symmetry. See Appendix A for a proof.

Proposition 1 (One step equivariance implies n-step equivariance) *If the one-step probabilistic forecasting model f_θ is G-equivariant, then the probability distribution $p_\theta(x_{t+1:t+k}|x_{1:t}, \mathbf{e})$ is invariant as in Equation 2.*

In order to enforce $\mathrm{SO}(2)$-equivariance for f_θ, the following proposition describes how the mean and covariance matrix for a 2-D Gaussian transforms under a rotation of $\mathbb{R}^2$.

Proposition 2 (SO(2) equivariance of multivariate Gaussian) *Given multivariate normal distribution $\mathcal{N}(\mu, \Sigma)$ over $\mathbb{R}^2$ with probability density function $p_{\mu,\Sigma}$ and $g \in \mathrm{SO}(2)$, then $\mathcal{N}(g\mu, g\Sigma g^T)$ is also a multivariate normal distribution and $p_{g\mu, g\Sigma g^T}(v) = p_{\mu,\Sigma}(g^{-1}v)$ for all $v \in \mathbb{R}^2$.*

To ensure the covariance matrix of f_θ is positive-definite and symmetric, i.e. $\Sigma \in \mathrm{PosDefSym}_2(\mathbb{R})$, we make use of the following fact:

Proposition 3 (Equivariant maps constructing positive-definite symmetric matrices) *The map*

$$\varphi \colon \mathrm{GL}_2(\mathbb{R}) \to \mathrm{PosDefSym}_2(\mathbb{R}), \quad M \mapsto MM^T \tag{3}$$

is surjective and equivariant. That is, for $g \in \mathrm{SO}(2)$ we have

$$\varphi(gM) = g\varphi(M)g^T. \tag{4}$$

Moreover, φ admits a one-sided inverse which is also equivariant,

$$\psi \colon \mathrm{PosDefSym}_2(\mathbb{R}) \to \mathrm{GL}_2(\mathbb{R}), \quad \Sigma \mapsto Q\Lambda^{\frac{1}{2}} \tag{5}$$

where $Q\Lambda Q^T$ is the eigendecomposition of Σ and Q is orthogonal. Together, $\varphi(\psi(\Sigma)) = \Sigma$.

As a consequence of Proposition 2 and Proposition 3, we can ensure that the predicted distribution transforms correctly under equivariance by (1) predicting an intermediate matrix M for covariance, and (2) constraining $\tilde{f}_\theta$ to be equivariant with respect to the action in Equation 4.

$$\mu_{j+1}, M_{j+1} = \tilde{f}_\theta(\mu_{j-k:j}, M_{j-k:j}, \mathbf{e}), \ \Sigma_{j+1} = M_{j+1}M_{j+1}^T \tag{6}$$

In this case, $\mathrm{SO}(2)$ acts by transforming the the columns of M independently as vectors in $\mathbb{R}^2$. Thus the data $(\mu_j^{(i)}, \Sigma_j^{(i)})$ for each agent and time step is comprised of 3 copies of the standard representation ρ_1 as defined in Section 3. Given this $\mathrm{SO}(2)$ action, we can enforce $\mathrm{SO}(2)$-equivariance in the neural network $\tilde{f}_\theta$. Implementation details of of the equivariant layers are provided in Appendix B.

4.3. Dynamics Integration (dyna)

Instead of predicting the position directly, PECCO outputs a Gaussian distribution over *velocity* as $\mathcal{N}(\mu_{v,j}, \Sigma_{v,j})$. More specifically, it predicts $(\mu_{v,j}, M_{v,j})$ at each time step and the covariance matrix $\Sigma_{v,j}$ is calculated as in Proposition 3. However, we want to obtain the uncertainty over *position* as $\Sigma_{x,j}$ and perform autoregressive forecasting. We leverage dynamics integration to propagate the uncertainty from velocity to position.

Assuming that all agents in the system can be approximated as linear discrete time dynamics $x_{j+1} = x_j + \Delta t \cdot v_j$, we can obtain the uncertainty of predicted position $\Sigma_{x,j+1}$ by

$$\Sigma_{x,j+1} = \Sigma_{x,j} + (\Delta t)^2 \Sigma_{v,j} + 2\Delta t \cdot \mathrm{cov}(x_j, v_j).$$

We assume that the cross covariance matrix $\text{cov}(x_j, v_j)$ is zero for simplicity following previous works (Salzmann et al., 2020). During training, gradients are calculated after entire trajectory is predicted autoregressively. A consequence of the additive uncertainty setup is that it enforces the uncertainty to grow monotonically over time, creating a "cone of uncertainty".

5. Experiments

We show that our model produces accurate and more calibrated probabilistic forecast compared to baseline models on one synthetic and two real world trajectory prediction datasets: interacting particles, autonomous vehicle, and pedestrian movement.

5.1. Baselines

- `LSTM-NLL` (variation of Alahi et al. (2016)): An encoder-decoder LSTM model that predicts the mean and variance of a Gaussian distribution, optimizing likelihood of data. We also train a version with random rotation data augmentation `LSTM-aug`.

- `CtsConv` (Ummenhofer et al., 2019): Continuous convolution over point cloud data for trajectory prediction, a non-equivariant counterpart of PECCO. `CtsConv-aug` is trained with a data augmentation step where we randomly rotate the scenes.

- `Multiple Futures Prediction (MFP)` (Tang and Salakhutdinov, 2019): A encoder-decoder model for multimodal probabilistic trajectory forecasts.

- `Trajectron++` (Salzmann et al., 2020): State-of-the-art probabilistic trajectory prediction model with graph representation of agent interactions and conditional VAE architecture.

5.2. Evaluation Metrics.

- *Minimum Average/Final Displacement Error* (minADE$_6$, minFDE$_6$): average l_2 displacement error over k steps, or average displacement of the final step, between predicted and ground truth trajectories. We report the minimum over 6 samples for probabilistic models.

- *Negative Log Likelihood* (NLL): NLL of ground truth trajectories under predicted distributions.

- *Energy Score* (ES) (Gneiting and Raftery, 2007): a proper scoring rule to measure calibration and sharpness of the predicted distribution P. The energy score for a distribution P and the ground truth data x is defined as: $\text{ES}(P, x) = E_{X \sim P}\|X - x\| - \frac{1}{2}E_{X, X' \sim P}\|X - X'\|$. Here X and X' are independent samples from P.

- *Coverage*: The empirical estimate of probability of the true value lying in the predicted interval, defined in equation 1. We report the coverage of 90% quantile of the predicted Gaussian. The prediction is more *calibrated* if the coverage is closer to 90%.

5.3. Datasets.

The particle dataset is a synthetic dataset of 5 particles interacting in spring dynamics (Kipf et al., 2018) with dynamics noise. The models predict 20 time steps given 30 steps as input. The Argoverse autonomous vehicle motion forecasting (Chang et al., 2019) is a widely used vehicle trajectory prediction benchmark. The task is to predict 3 second trajectories based on all vehicle history in the past 2 seconds recorded at 10Hz. TrajNet++ (Sadeghian et al., 2018) is a popular pedestrian

Model	minADE$_6$ ↓	minFDE$_6$ ↓	NLL↓	ES ↓	Cov@1s(%)	Cov@2s	Cov@3s
			Argoverse				
LSTM-NLL	1.64 ± .05	4.17 ± .10	3.07 ± .08	2.31 ± .54	8.8 ± 0.7	8.5 ± 0.7	7.0 ± 0.8
LSTM-NLL-aug	1.61 ± .02	4.15 ± .08	2.78 ± .03	1.99 ± .46	10.1 ± 1.5	10.5 ± 1.0	9.8 ± 1.9
CtsConv-NLL	1.74 ± .03	4.43 ± .06	29.1 ± 2.2	6.71 ± .70	6.3 ± 2.2	0.02 ± .01	0.01 ± .01
CtsConv-NLL-aug	1.66 ± .02	4.23 ± .06	11.81 ± .01	5.10 ± .35	11.9 ± 2.1	1.7 ± 0.5	0.02 ± .01
Trajectron++	1.83 ± .02	3.85 ± .07	**2.48** ± .27	3.92 ± .61	45.5 ± 5.3	37.6 ± 3.2	34.9 ± 2.5
MFP	1.53 ± .04	3.77 ± .06	3.56 ± .02	2.33 ± .21	51.3 ± 5.1	33.0 ± 4.9	8.3 ± 4.8
PECCO	**1.39** ± .02	**3.41** ± .03	4.26 ± 0.1	**1.54** ± .16	**74.9** ± 0.6	**78.6** ± 2.8	**84.5** ± 2.9
			TrajNet++				
LSTM-NLL-aug	0.85 ± .02	1.64 ± .03	2.78 ± .02	-0.28 ± .09	29.0 ± 4.3	23.2 ± 4.2	23.7 ± 3.9
CtsCov-NLL	1.08 ± .02	2.36 ± .09	5.33 ± .08	1.67 ± .13	43.8 ± 10.6	20.7 ± 5.2	12.2 ± 6.7
CtsCov-NLL-aug	0.92 ± .01	1.76 ± .03	6.74 ± .21	1.42 ± .11	62.1 ± 3.3	36.3 ± 4.9	34.1 ± 5.8
Trajectron++	1.14 ± .03	2.31 ± .05	2.83 ± .12	0.98 ± .17	50.2 ± 2.2	45.8 ± 3.5	32.9 ± 3.5
MFP	0.85 ± .02	1.70 ± .04	**2.20** ± .04	0.67 ± .08	79.1 ± 4.3	32.5 ± 3.1	22.8 ± 3.2
PECCO	**0.59** ± .12	**1.06** ± .17	2.37 ± .04	**-0.73** ± .10	**80.8** ± 4.5	**85.9** ± 2.3	**94.5** ± 3.0

Table 2: Performance comparison on benchmark datasets Argoverse and TrajNet++. Cov@ks(%) refers to coverage at the k second mark; prediction is more *calibrated* if closer to 90%. PECCO is more accurate and calibrated compared to non-equivariant baseline models.

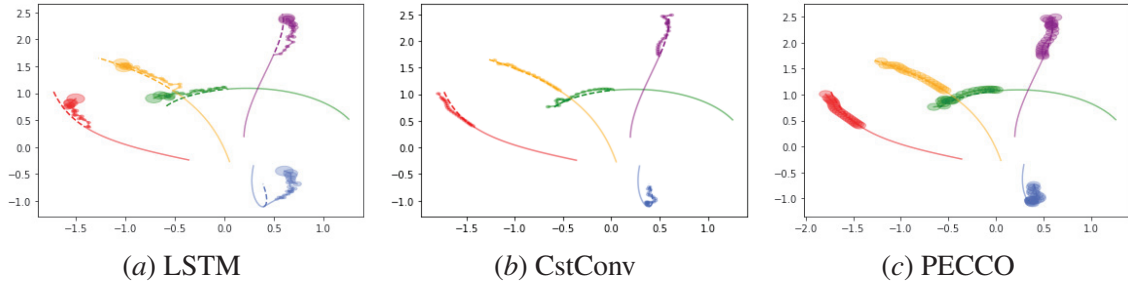(*a*) LSTM   (*b*) CstConv   (*c*) PECCO

Figure 3: Comparison of prediction results between baselines. The solid lines are input timesteps to the models, the dotted lines ground truth, and the circles the 90% confidence regions. We can see that PECCO achieves accurate results while maintaining good coverage.

trajectory benchmark with a focus on agent-agent interaction scenarios. The task is to predict 12 time steps for agents given 9. We refer the reader to Appendix D.1 for data and training details.

5.4. Experimental Results

Particles dataset. We present experimental result for the synthetic particle dataset in Table 1. We visualize a test sample in Figure 3 to qualitatively illustrate PECCO's improved accuracy and calibration.

Argoverse and TrajNet++. Table 2 presents experimental results on two benchmark datasets. PECCO achieves better regression accuracy compared to non-equivariant baseline in terms of minADE and minFDE, with a notable 9% improve-

Model	MSE ↓	NLL↓	ES ↓
LSTM	.016	-1.61	1.041
CtsConv	.010	-0.81	0.667
PECCO	**.004**	**-0.83**	**0.467**

Table 1: PECCO outperforms baseline models in all metrics on the synthetic particles dataset.

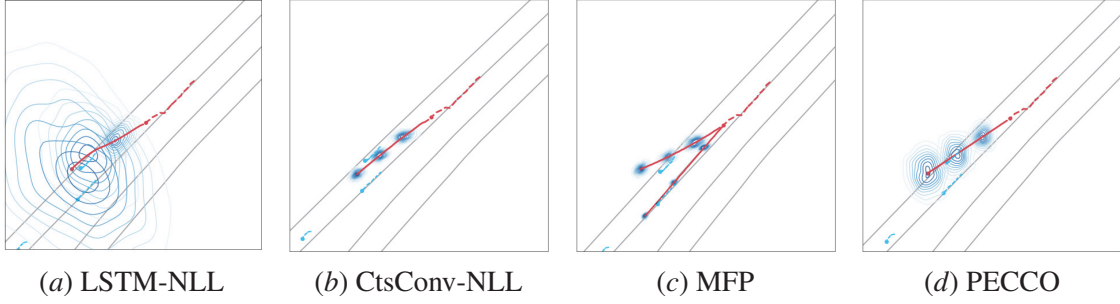(a) LSTM-NLL   (b) CtsConv-NLL   (c) MFP   (d) PECCO

Figure 4: Comparison of uncertainty predicted at a lane change. Red trajectories are the agent of interest in the same scenario. Note how the LSTM predicted uncertainty explodes after a few time steps, while CtsConv and MFP has overconfident distributions. PECCO is able to model possibility of both staying and lane change.
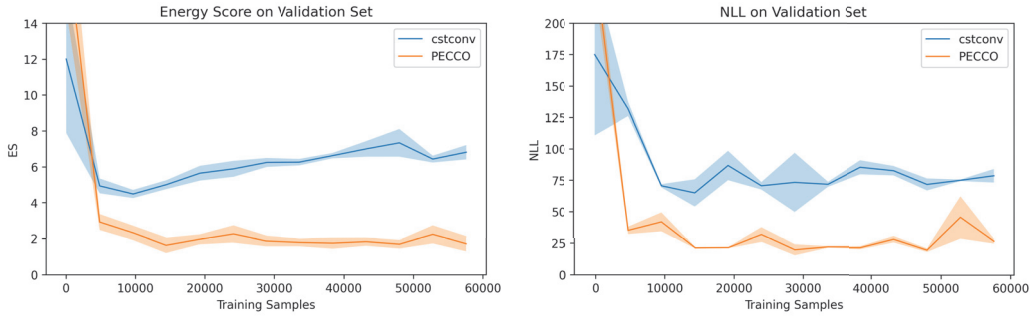

Figure 5: With equivariance, PECCO is able to achieve better energy score and NLL with fewer data samples, compared to its non-equivariant counterpart.

ment in minADE over the the best performing baseline, MFP.

PECCO's improved probability coverage allows for more diverse sampling and hence can produce trajectories closer to ground truth.

PECCO's probabilistic predictions are able to achieve consistently better coverage compared to other methods whose coverage deteriorates over time. Comparing LSTM-NLL and CtsConv-NLL with their augmented counterparts, LSTM-NLL-aug and CtsConv-NLL-aug, we can see that data augmentation through rotation improves both accuracy and calibration. PECCO leverages this symmetry to improve accuracy, maintain good calibration, and converge faster (Figure 5).

Figure 4 visualizes a typical situation to illustrate this difference. We plot the predicted distribution at 10, 20, and 30 time steps of prediction (1 time step is 0.1 seconds). We can see the probable region predicted by LSTM-NLL explodes at timesteps 20 and 30, whereas CtsConv-NLL and MFP tend to be overconfident in their predictions. PECCO is able to predict a Gaussian that covers both cases of staying in the lane and changing to the right lane.

5.5. Model Analysis

Comparison with data augmentation and cannonicalization. Data augmentation and cannonicalization are popular methods to implicitly exploit symmetry in trajectory data. In Table 4 we

Model	minADE$_6$ ↓	minFDE$_6$ ↓	MRS ↓	Cov@1s(%)	Cov@2s	Cov@3s
Conformal LSTM	2.45 ± 0.09	4.68 ± 0.15	198.1 ± 12.0	90.1 ± 0.1	92.7 ± 0.1	92.8 ± 0.1
Conformal ECCO	1.96 ± 0.06	4.32 ± 0.10	220.9 ± 8.1	**90.0** ± 0.1	**90.1** ± 0.1	**90.0** ± 0.1
PECCO	**1.39** ± .02	**3.41** ± 0.03	**8.52** ± 0.16	74.9 ± 0.2	93.6 ± 0.8	92.5 ± 0.9

Table 3: Comparison with conformal prediction methods on Argoverse dataset. PECCO produces a parametric distribution with a tighter confidence region (small MRS), whereby achieving better regression accuracy while maintaining competitive coverage.

compare PECCO to its nonequivariant counterparts with the addition of data augmentation and canonicalization. Equivariance dramatically improves prediction performance in all aspects, especially calibration.

Dynamics Integration Ablation. Dynamic integration (dyna) introduced in Section 4.3 enforces the uncertainty to grow monotonically over time. As an ablative study, Table 5 show that PECCO with dynamic integration has much better calibration compared to if without; for the Argoverse dataset, the improved calibration also informs better performance on minADE/FDE.

	minA/FDE$_6$	NLL	Cov(%)
CtsConv	1.74 / 4.43	29.13	2.2
+cannon	1.66 / 4.28	17.46	4.5
+aug	1.67 / 4.23	11.81	17.1
equivariant	**1.39 / 3.41**	**4.26**	**87.5**

Table 4: Data Augmentation Comparison on the Argoverse Dataset.

Comparison with Conformal Prediction. We compare to two conformal prediction baselines (Stankevičiūtė et al., 2021) (Appendix C.3 for details) in Table 3. Conformal methods achieve guaranteed ≥ 90% coverage, but suffer in prediction accuracy due to having to split training data for calibration. Since conformal prediction does not output a distribution, we use another proper scoring rule Mean Regional Score (2d extension of mean interval score in Gneiting and Raftery (2007), see Appendix B) as metric. The conformal regions are much larger (higher MRS), which is less desirable for downstream decision making tasks.

6. Conclusion

In this work we propose Probabilistic Equivariant Continuous Convolution (PECCO), a novel multi-agent probabilistic prediction method for improving uncertainty quantification. We design an equivariant neural network under which the predicted distributions transform correspondingly as inputs are transformed. We introduce the Energy Score metric to bring attention to the calibration of multivariate probabilistic forecasts. By leveraging equivariance, PECCO produces more accurate and calibrated probabilistic forecasts compared to existing methods on both synthetic and real-world datasets.

Argoverse	minA/FDE$_6$	NLL	Cov(%)
no-dyna	1.52 / 3.76	9.72	38.6
dyna	**1.39 / 3.41**	**4.26**	**87.5**
Pedestrian	minA/FDE$_6$	NLL	Cov(%)
no-dyna	**0.72** / 2.12	4.71	39.6
dyna	0.73 / **1.98**	**2.37**	**83.7**

Table 5: Dynamics Integration (dyna) Ablation: using dyna encourages the uncertainty to grow over time and improves coverage.

Acknowledgments

References

Alexandre Alahi, Kratarth Goel, Vignesh Ramanathan, Alexandre Robicquet, Li Fei-Fei, and Silvio Savarese. Social lstm: Human trajectory prediction in crowded spaces. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016.

Erkao Bao and Linqi Song. Equivariant neural networks and equivarification. *arXiv preprint arXiv:1906.07172*, 2019.

Benjamin Bloem-Reddy and Yee Whye Teh. Probabilistic symmetries and invariant neural networks. *Journal of Machine Learning Research*, 21(90), 2020.

Monimoy Bujarbaruah, Xiaojing Zhang, Marko Tanaskovic, and Francesco Borrelli. Adaptive mpc under time varying uncertainty: Robust and stochastic. *arXiv preprint arXiv:1909.13473*, 2019.

Yuning Chai, Benjamin Sapp, Mayank Bansal, and Dragomir Anguelov. Multipath: Multiple probabilistic anchor trajectory hypotheses for behavior prediction. *arXiv preprint arXiv:1910.05449*, 2019.

Ming-Fang Chang, John W Lambert, Patsorn Sangkloy, Jagjeet Singh, Slawomir Bak, Andrew Hartnett, De Wang, Peter Carr, Simon Lucey, Deva Ramanan, and James Hays. Argoverse: 3d tracking and forecasting with rich maps. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

Nachiket Deo and Mohan M Trivedi. Convolutional social pooling for vehicle trajectory prediction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2018.

Fabian B Fuchs, Daniel E Worrall, Volker Fischer, and Max Welling. SE(3)-transformers: 3D roto-translation equivariant attention networks. *arXiv preprint arXiv:2006.10503*, 2020.

Jiyang Gao, Chen Sun, Hang Zhao, Yi Shen, Dragomir Anguelov, Congcong Li, and Cordelia Schmid. Vectornet: Encoding hd maps and agent dynamics from vectorized representation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020.

Tilmann Gneiting and Adrian E Raftery. Strictly proper scoring rules, prediction, and estimation. *Journal of the American statistical Association*, 102(477), 2007.

Junru Gu, Chen Sun, and Hang Zhao. Densetnt: End-to-end trajectory prediction from dense goal sets. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. On calibration of modern neural networks. In *International Conference on Machine Learning*. PMLR, 2017.

Agrim Gupta, Justin Johnson, Li Fei-Fei, Silvio Savarese, and Alexandre Alahi. Social gan: Socially acceptable trajectories with generative adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

Boris Ivanovic and Marco Pavone. Rethinking trajectory forecasting evaluation. *arXiv preprint arXiv:2107.10297*, 2021.

Michael I. Jordan, editor. *Prediction with Gaussian Processes: From Linear Regression to Linear Prediction and Beyond*. The MIT Press, 1998.

Rudolph Emil Kalman. A new approach to linear filtering and prediction problems. *Journal of Basic Engineering*, 82(1), 1960.

Thomas Kipf, Ethan Fetaya, Kuan-Chieh Wang, Max Welling, and Richard Zemel. Neural relational inference for interacting systems. In *International Conference on Machine Learning*, pages 2688–2697. PMLR, 2018.

Jonas Köhler, Leon Klein, and Frank Noé. Equivariant flows: exact likelihood generative learning for symmetric densities. In *International Conference on Machine Learning*. PMLR, 2020.

Namhoon Lee, Wongun Choi, Paul Vernaza, Christopher B Choy, Philip HS Torr, and Manmohan Chandraker. Desire: Distant future prediction in dynamic scenes with interacting agents. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

Ming Liang, Bin Yang, Rui Hu, Yun Chen, Renjie Liao, Song Feng, and Raquel Urtasun. Learning lane graph representations for motion forecasting. In *European Conference on Computer Vision*. Springer, 2020.

Yukai Liu, Rose Yu, Stephan Zheng, Eric Zhan, and Yisong Yue. Naomi: Non-autoregressive multiresolution sequence imputation. *Advances in Neural Information Processing Systems*, 2019.

Rachel Luo, Shengjia Zhao, Jonathan Kuck, Boris Ivanovic, Silvio Savarese, Edward Schmerling, and Marco Pavone. Sample-efficient safety assurances using conformal prediction. *arXiv preprint arXiv:2109.14082*, 2021.

Chris J Ostafew, Angela P Schoellig, Timothy D Barfoot, and Jack Collier. Learning-based nonlinear model predictive control to improve vision-based mobile robot path tracking. *Journal of Field Robotics*, 33(1), 2016.

T Mitchell Roddenberry, Nicholas Glaze, and Santiago Segarra. Principled simplicial neural networks for trajectory prediction. In *International Conference on Machine Learning*. PMLR, 2021.

Andrey Rudenko, Luigi Palmieri, Michael Herman, Kris M Kitani, Dariu M Gavrila, and Kai O Arras. Human motion trajectory prediction: a survey. *The International Journal of Robotics Research*, 39(8), 2020.

Amir Sadeghian, Vineet Kosaraju, Agrim Gupta, Silvio Savarese, and Alexandre Alahi. Trajnet: Towards a benchmark for human trajectory prediction. *arXiv preprint*, 2018.

Amir Sadeghian, Vineet Kosaraju, Ali Sadeghian, Noriaki Hirose, Hamid Rezatofighi, and Silvio Savarese. Sophie: An attentive gan for predicting paths compliant to social and physical constraints. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019.

David Salinas, Michael Bohlke-Schneider, Laurent Callot, Roberto Medico, and Jan Gasthaus. High-dimensional multivariate forecasting with low-rank gaussian copula processes. *Advances in neural information processing systems*, 32, 2019.

David Salinas, Valentin Flunkert, Jan Gasthaus, and Tim Januschowski. Deepar: Probabilistic forecasting with autoregressive recurrent networks. *International Journal of Forecasting*, 36(3): 1181–1191, 2020.

Tim Salzmann, Boris Ivanovic, Punarjay Chakravarty, and Marco Pavone. Trajectron++: Dynamically-feasible trajectory forecasting with heterogeneous data. In *ECCV*. Springer, 2020.

Victor Garcia Satorras, Emiel Hoogeboom, Fabian Bernd Fuchs, Ingmar Posner, and Max Welling. E (n) equivariant normalizing flows. In *Thirty-Fifth Conference on Neural Information Processing Systems*, 2021.

Wilko Schwarting, Javier Alonso-Mora, and Daniela Rus. Planning and decision-making for autonomous vehicles. *Annual Review of Control, Robotics, and Autonomous Systems*, 1, 2018.

Glenn Shafer and Vladimir Vovk. A tutorial on conformal prediction. *Journal of Machine Learning Research*, 9(3), 2008.

Mlnoru Slotani. Tolerance regions for a multivariate normal population. *Annals of the Institute of Statistical Mathematics*, 16(1), 1964.

Kihyuk Sohn, Honglak Lee, and Xinchen Yan. Learning structured output representation using deep conditional generative models. *Advances in neural information processing systems*, 28, 2015.

Kamilė Stankevičiūtė, Ahmed Alaa, and Mihaela van der Schaar. Conformal time-series forecasting. In *Advances in Neural Information Processing Systems*, 2021.

Jennifer J Sun, Tomomi Karigo, Dipam Chakraborty, Sharada Mohanty, Benjamin Wild, Quan Sun, Chen Chen, David Anderson, Pietro Perona, Yisong Yue, et al. The multi-agent behavior dataset: Mouse dyadic social interactions. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 1)*, 2021.

Yichuan Charlie Tang and Ruslan Salakhutdinov. Multiple futures prediction. *Advances in Neural Information Processing Systems*, 2019.

Benjamin Ummenhofer, Lukas Prantl, Nils Thuerey, and Vladlen Koltun. Lagrangian fluid simulation with continuous convolutions. In *International Conference on Learning Representations*, 2019.

Elise van der Pol, Daniel Worrall, Herke van Hoof, Frans Oliehoek, and Max Welling. Mdp homomorphic networks: Group symmetries in reinforcement learning. *Advances in Neural Information Processing Systems*, 33, 2020.

Robin Walters, Jinxi Li, and Rose Yu. Trajectory prediction using equivariant continuous convolution. *International Conference on Learning Representations*, 2021.

Rui Wang, Robin Walters, and Rose Yu. Incorporating symmetry into deep dynamics models for improved generalization. *International Conference on Learning Representations*, 2021.

Shenlong Wang, Simon Suo, Wei-Chiu Ma, Andrei Pokrovsky, and Raquel Urtasun. Deep parametric continuous convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

Maurice Weiler and Gabriele Cesa. General E(2)-equivariant steerable CNNs. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

Daniel E Worrall, Stephan J Garbin, Daniyar Turmukhambetov, and Gabriel J Brostow. Harmonic networks: Deep translation and rotation equivariance. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

Dongxia Wu, Liyao Gao, Xinyue Xiong, Matteo Chinazzi, Alessandro Vespignani, Yi-An Ma, and Rose Yu. Quantifying uncertainty in deep spatiotemporal forecasting. *ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, 2021.