THE KINEMATIC SIMILARITY PRINCIPLE: A SHARED KINEMATIC DISTRIBUTION ACROSS MOBILITY DOMAINS ENABLES ZERO-SHOT TRAJECTORY PREDICTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Existing trajectory predictors are specialized to a single mobility domain, such as pedestrians, vehicles, vessels, or aircraft. Zero-shot transfer has so far been studied within one domain, where the target is a new city, dataset, or task, while the mobility domain remains fixed. When the mobility domain is held out, existing predictors do not consistently outperform simple constant-velocity extrapolation. This is because they learn the scale and context of the domain they are trained on rather than the structure of motion that transcends domains. We show that such a shared motion structure indeed exists in the form of a shared kinematic distribution of short-horizon motion. Once such motion is expressed at the scale of its domain, domains share a distribution of future displacement given past kinematics. We call this regularity the *kinematic similarity principle*. A predictor that targets this shared motion can therefore be trained on data from some domains and applied to a held-out domain. We propose KINETRAJ, a predictor that exploits this principle. It includes (i) a scale-invariant kinematic encoder that expresses windows at the characteristic scale of its domain; (ii) a domain-agnostic backbone that takes only the resulting normalized representation as input, employing no domain labels or absolute scales; and (iii) a so-called KineMix decoder that combines closed-form motion regimes through a learned gate and residual to rescale predictions to the scale of a domain. On a leave-one-domain-out benchmark encompassing six datasets spanning five mobility domains, KINETRAJ, with 0.59M parameters, reduces best-of-20 ADE by 18% and FDE by 22% relative to the strongest domain-specific predictor trained under the same protocol, and it matches or exceeds trajectory and time-series foundation models applied zero-shot. Code is available at https://anonymous.4open.science/r/kinetraj_code-0D48.

1 INTRODUCTION

Trajectory prediction underpins autonomous driving (Chang et al., 2019; Ettinger et al., 2021; Shi et al., 2022), pedestrian safety (Alahi et al., 2016; Gupta et al., 2018), maritime surveillance (Nguyen & Fablet, 2024), and location-based services over human GPS traces (Zhu et al., 2025; Yang et al., 2025). Each application has developed its own predictors, each trained and evaluated inside a single mobility domain. A new sensing platform or class of agent therefore starts without a predictor, and collecting and curating enough of its own trajectories to train one takes time that a deployment may not have (Xu et al., 2022; Ivanovic et al., 2023). Millions of trajectories already exist in other mobility domains. We ask *whether they carry structure that a new domain can use, and how much a predictor trained on them alone gives up against one trained in the new domain*.

Existing zero-shot and transfer studies keep the mobility domain fixed. Cross-dataset and cross-region transfer for vehicles retains road geometry and comparable motion constraints (Feng et al., 2024; Lin et al., 2024; Wei et al., 2025), trajectory foundation models pretrained on worldwide vehicle GPS traces or on human motion still forecast the class of agent they were pretrained on (Zhu et al., 2025; Gao et al., 2027), and pedestrian domain-generalization methods change scenes or cameras but still forecast pedestrians (Huang et al., 2022; Dong et al., 2025; Li et al., 2024). In all

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

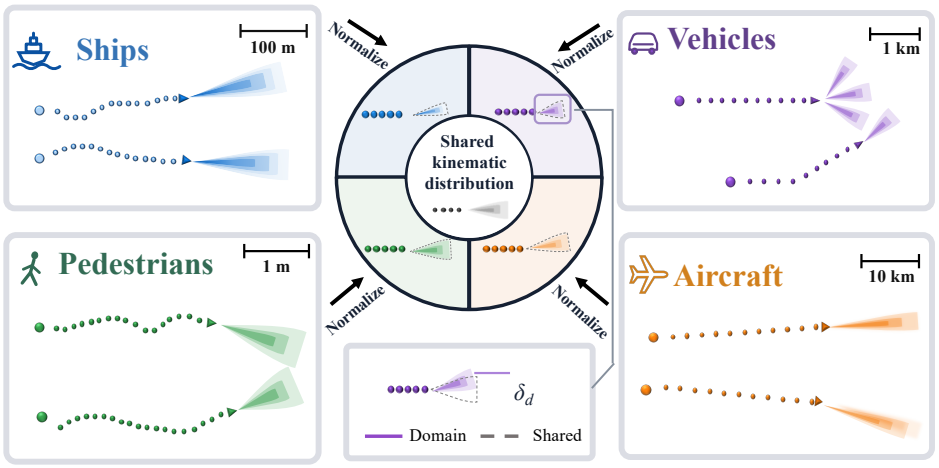


Figure 1: The kinematic similarity principle. Outer panels: four mobility domains at their native scales; each trajectory terminates in its distribution of likely futures. Ring: the same distributions after normalization by each domain's characteristic scale, superimposed on the shared distribution (dashed), which the hub shows alone. Bottom box: the vehicle distribution enlarged; its deviation from the shared distribution is the departure δ_d. All distributions are schematic.

of these the agent type, sampling conventions, and motion scale stay comparable (Feng et al., 2024; Zhu et al., 2025; Li et al., 2024; Westny et al., 2026). Time-series foundation models (Shi et al., 2025; Das et al., 2024; Ansari et al., 2024) forecast any series zero-shot but see a trajectory as a generic sequence, not as motion. None of them asks what happens when an entire mobility domain is absent from training, with a different agent type, sensing process, sampling interval, and scale.

We study this setting on the ships, vehicles, pedestrians, and aircraft of Figure 1, whose motion scales span four orders of magnitude and on which domain-specific predictors (Alahi et al., 2016; Salzmann et al., 2020) do not consistently improve over constant-velocity extrapolation. The question is therefore what, if anything, transfers across these domains. Our central finding is that it is a *shared kinematic distribution of short-horizon motion*. Despite their scales, all four domains share inertia, bounded acceleration, and locally smooth heading changes, and once each is expressed in its own characteristic scale, the conditional distribution of future displacement given past kinematics, the domain's *kinematic distribution*, agrees across domains up to a *departure* δ_d that measures what it does not share. We term this regularity the *kinematic similarity principle*. Section 3 formalizes it and shows that the shared distribution sets the limit of zero-shot prediction. The predictor that is optimal on the sources, applied in normalized coordinates to a new domain without its data, falls short of a predictor trained on that domain by exactly its departure δ_d.

KINETRAJ realizes the principle with three components: (i) a scale-invariant kinematic encoder that normalizes each window by the characteristic scales of its domain, statistics of that domain's data, and supplies the coordinates in which the shared distribution lives; (ii) a domain-agnostic backbone that operates only on this normalized representation and receives no domain label or absolute scale; and (iii) the KineMix decoder, which realizes the shared distribution as a mixture over a few motion *regimes*, keeping the regime set as closed-form physics anchors, learning the regime posterior as a gate and the within-regime deviation as residual modes, and rescaling the prediction to the domain's scale. On the six datasets, KINETRAJ, with 0.59M parameters, is the best method on five and within 2% of the strongest foundation model on the sixth, reduces best-of-20 ADE by 18% and FDE by 22% relative to the strongest domain-specific predictor trained under the same protocol, and matches or exceeds trajectory and time-series foundation models applied zero-shot; the advantage persists on the aircraft dataset, whose characteristic speed is 14 times the largest training scale.

Our contributions can be summarized as:

- We state the kinematic similarity principle, prove that a single normalized predictor falls short of in-domain training on any domain by exactly that domain's departure δ_d, which sets the limit of zero-shot prediction, and derive four implications, on coverage, scale extrapolation, departure, and absolute scale, that the experiments verify.

- We design KINETRAJ, which realizes the shared distribution as a mixture over closed-form motion regimes with a scale-invariant encoder, a domain-agnostic backbone, and the KineMix decoder.
- We formulate a leave-one-domain-out benchmark over six datasets spanning five mobility domains, including an off-scale aircraft domain, evaluate against three groups of baselines, and verify the principle's implications on it.

The rest of the paper is organized as follows. Section 2 reviews the domain-specific predictors, transfer studies, and foundation models that this work builds on and departs from. Section 3 states the kinematic similarity principle, derives the excess risk of zero-shot prediction from it, and lists four implications that experiments can test. Section 4 presents KINETRAJ, whose three components realize the principle. Section 5 describes the leave-one-domain-out benchmark, compares KINETRAJ with three groups of baselines, and verifies the four implications, and Section 6 concludes.

2 RELATED WORK

Trajectory predictors have been developed for one mobility domain at a time, and transfer studies among them have moved between datasets without leaving a domain, so foundation models are the first candidates for a prior that crosses domains. We review these three lines of work, then the physics mixtures and equivariant models that KINETRAJ builds on, and finally the similarity theory that gives the principle its name.

Domain-specific predictors. Pedestrian models pool social interactions (Alahi et al., 2016; Gupta et al., 2018; Yuan et al., 2021), driving models add high-definition (HD) map context and lane graphs (Shi et al., 2022; Salzmann et al., 2020), maritime forecasting uses AIS-specific encodings (Nguyen & Fablet, 2024), and multi-class road-user predictors (Rainbow et al., 2021) observe every class during training. Social pooling and scene graphs presuppose a shared scene that a vessel and a pedestrian do not have, so applying these models to a held-out domain means removing the inputs that define them, leaving the coordinate-only cores we compare against.

Zero-shot transfer within a mobility domain. Transfer studies change the city, dataset, camera, region, or task while keeping the mobility domain fixed: UniTraj for vehicles (Feng et al., 2024) reports large drops under cross-dataset transfer, TrajFM (Lin et al., 2024) and TransferTraj (Wei et al., 2025) target vehicle region and task transfer with road context, the universal trajectory model UniTraj (Zhu et al., 2025) pretrains on worldwide vehicle GPS traces, and pedestrian domain generalization holds out new pedestrian scenes by feature alignment (Huang et al., 2022; Dong et al., 2025), meta-learning (Li et al., 2024), or a constant-velocity prior with learned residuals (Huang et al., 2023). Frenet normalization removes road-geometry differences in driving (Ye et al., 2023), and a 24-benchmark study finds even within-family transfer difficult (Westny et al., 2026). In all of these the agent type, sampling conventions, and motion scale stay comparable (Appendix C).

Foundation models. General time-series models such as Time-MoE (Shi et al., 2025), TimesFM (Das et al., 2024), and Chronos (Ansari et al., 2024), and the trajectory and mobility foundation models surveyed by Yang et al. (2025), promise priors that transfer anywhere; on aggregate mobility series they deliver (Yanes-Pulido & Rodrigues, 2026; Graser, 2025), and spatio-temporal foundation models (Yuan et al., 2024; Huang et al., 2025; Li et al., 2025) generalize across cities but forecast aggregate traffic on grids or road graphs rather than individual trajectories.

Physics mixtures and equivariance. The interacting multiple model filter (Blom & Bar-Shalom, 1988) switches between constant-velocity, constant-acceleration, and coordinated-turn models (Li & Jilkov, 2003), and learned predictors embed kinematics as a feasibility layer (Cui et al., 2020; Salzmann et al., 2020) or as a constant-velocity prior with learned residuals (Huang et al., 2023). KineMix (Section 4) differs in that its anchors live in scale-normalized coordinates, so one mixture serves every domain. EqMotion (Xu et al., 2023) is Euclidean-equivariant; we pursue equivariance to the scaling of velocity and acceleration, the axis along which mobility domains differ.

Rescaling and similarity. Human mobility collapses onto one distribution after rescaling by each person's radius of gyration (González et al., 2008), and mammals of different sizes move alike at equal Froude number (Alexander & Jayes, 1983). In similarity theory (Buckingham, 1914) two motions are kinematically similar when corresponding velocities differ by a single factor and dynamically similar when every dimensionless group agrees; the principle asserts the former, in distri-

bution, so that a few characteristic scales per domain suffice over a short horizon is a finding, not a corollary. To our knowledge, no such principle has been stated for trajectory prediction.

3 The Kinematic Similarity Principle: What Transfers across Mobility Domains

This section defines the kinematic similarity principle, derives from it the limit of zero-shot prediction, set by the held-out domain's departure from the shared distribution, and draws from the principle and the bound four implications that the experiments can verify.

3.1 The Kinematic Similarity Principle

An observed window of an agent's motion is summarized by a kinematic description $\mathbf{k} = (\mathbf{k}_{\mathrm{m}}, \mathbf{k}_{\mathrm{i}})$ with two groups of channels, the *magnitude* channels $\mathbf{k}_{\mathrm{m}}$, which carry a physical unit such as a speed, and the dimensionless *scale-invariant* channels $\mathbf{k}_{\mathrm{i}}$, such as headings and ratios of two lengths. The prediction target $\mathbf{y}$ is the future displacement over a fixed short horizon of T steps from the last observed point, and for mobility domain d a window and its future are drawn jointly from $P_d(\mathbf{k}, \mathbf{y})$. A domain has characteristic scales s_d, one for each physical unit in the magnitude channels or in the target $\mathbf{y}$, the latter, the scale of the per-step displacement, denoted s_y, and the normalization divides each magnitude channel by the scale of its unit and leaves the scale-invariant channels unchanged,

$$\phi_{s_d}(\mathbf{k}) = \big(\mathbf{k}_{\mathrm{m}}/s_d,\ \mathbf{k}_{\mathrm{i}}\big), \qquad \mathbf{y}' = \mathbf{y}/s_y, \tag{1}$$

so that $\mathbf{k}' = \phi_{s_d}(\mathbf{k})$ and $\mathbf{y}'$ are dimensionless.

Definition 1 (Kinematic distribution)**.** For a mobility domain d with characteristic scales s_d, the *kinematic distribution* of d is the conditional distribution of normalized futures given normalized pasts,

$$Q_d(\mathbf{y}' \mid \mathbf{k}') \;=\; P_d\big(s_y^{-1}\mathbf{y} \,\big|\, \phi_{s_d}(\mathbf{k})\big)\,. \tag{2}$$

Definition 2 (Departure)**.** Let Q be a conditional distribution of $\mathbf{y}'$ given $\mathbf{k}'$ with conditional mean $g^\star(\mathbf{k}') = \mathbb{E}_Q[\mathbf{y}' \mid \mathbf{k}']$. The *departure* of domain d from Q is

$$\delta_d(Q) = \Big(\mathbb{E}_d\big\|g_d^\star(\mathbf{k}') - g^\star(\mathbf{k}')\big\|^2\Big)^{1/2}, \tag{3}$$

where $g_d^\star(\mathbf{k}') = \mathbb{E}_{Q_d}[\mathbf{y}' \mid \mathbf{k}']$ is the conditional mean of Q_d and $\mathbb{E}_d$ denotes expectation over the normalized pasts $\mathbf{k}' = \phi_{s_d}(\mathbf{k})$ of windows drawn from P_d.

Principle 1 (Kinematic similarity)**.** There is a conditional distribution Q of $\mathbf{y}'$ given $\mathbf{k}'$, the *shared distribution*, such that the kinematic distributions of mobility domains agree with Q up to their departures $\delta_d(Q)$. The exact case is $Q_d = Q$ for all d, for which $\delta_d(Q) = 0$.

Throughout, Q is the kinematic distribution of the pooled source domains, the distribution of $(\mathbf{k}', \mathbf{y}')$ obtained by normalizing each source domain by its own scales and pooling the results; $\delta_d(Q)$ is thus measured against the source set of the fold and written δ_d for short, a source domain against a pool that contains it and the held-out domain against one that does not. The principle concerns only what domains share; structure specific to a domain is available only to a predictor trained on its data, and δ_d bounds how close zero-shot prediction can come to in-domain training. The principle is confined to the short-horizon motion of a single agent; maps, social interaction, and intent lie outside it and are not claimed to transfer.

3.2 The Excess Risk of Zero-Shot Prediction

A zero-shot predictor is trained on source domains and applied to a domain outside that set, given only its characteristic scales. Under the kinematic similarity principle, the error of the predictor that is optimal on the sources is set by the held-out domain's departure from the shared distribution.

Proposition 1 (Optimal zero-shot error)**.** *Let Q be the kinematic distribution of the pooled source domains, with conditional mean $g^\star$, and let $d^\star$ be a domain outside the source set, with characteristic scales $s_{d^\star}$, step-displacement scale s_y, and departure $\delta_{d^\star}(Q)$. The predictor $\hat{\mathbf{y}}(\mathbf{k}) = s_y\, g^\star(\phi_{s_{d^\star}}(\mathbf{k}))$, which applies the conditional mean of Q to $d^\star$ in its own scales, satisfies*

$$\mathbb{E}_{P_{d^\star}}\big\|\hat{\mathbf{y}}(\mathbf{k}) - \mathbb{E}[\mathbf{y} \mid \mathbf{k}]\big\|^2 \;=\; s_y^2\,\delta_{d^\star}(Q)^2. \tag{4}$$

The reference $\mathbb{E}[\mathbf{y} \mid \mathbf{k}]$ is the Bayes predictor of $d^\star$, which training on the domain's own data approximates; because the normalization is invertible within a domain, it equals $s_y\, g^\star_{d^\star}(\phi_{s_{d^\star}}(\mathbf{k}))$, so the left side is the excess squared risk of the zero-shot predictor over the in-domain optimum. Any predictor of $\mathbf{k}'$ alone that is risk-optimal on the source pool coincides with $g^\star$ on the support of Q; a predictor estimated from finite source data adds its estimation error on the normalized inputs of $d^\star$, and a relative error ϵ in the supplied scales adds a term of order $|\epsilon|$ that enters a scale-equivariant predictor only through its learned corrections. Proposition 1 therefore bounds what zero-shot prediction can attain, since it fixes the excess risk of the source-optimal predictor over the in-domain optimum at exactly $s_y^2\delta^2_{d^\star}$, with the departure alone determining it; a predictor that does better on $d^\star$ is not one the sources can select. Appendix D gives the proof and extends the identity to source domains, learned maps, scale errors, and bounded losses such as the best-of-K error.

3.3 Verifiable Implications

The principle and Proposition 1 yield four implications that are experimentally verified. (i) A predictor trained in normalized coordinates improves multimodal coverage on a held-out domain, whereas predictors that bind their output magnitude to the training scale do not. (ii) The bound is stated in normalized units and does not involve the absolute scale of the target domain, so for a predictor that is scale-equivariant by construction (Section 4) the advantage should persist when the characteristic speed of the target lies far outside the training range. (iii) By Proposition 1, the excess error of zero-shot prediction over in-domain prediction should equal $s_y^2\delta_d^2$, so that a domain with a larger departure pays a larger excess, and within a domain whose windows mix transport modes the excess should concentrate on the mode that the sources share least. (iv) Within a domain the absolute speed of a window carries no information beyond $\mathbf{k}'$; across the sources it identifies the domain, so a predictor given it can at most substitute that domain's rule for the shared one, which differs by its departure (Corollary 1). Supplying it alters the in-range error by at most the source departures δ_d and cannot help outside the training range, where it lies beyond the training support.

The implications are experimentally verified in Section 5.3 and Appendix B.2.

4 KineTraj: Learning the Shared Kinematic Distribution

4.1 Framework Overview

Up to δ_d, the scales s_d are the only domain-specific quantities in the bound of Section 3, so a predictor that attains it must meet three requirements: it operates in normalized coordinates, it is a single function shared by every domain rather than one per domain, and it is scale-equivariant, so that an error in the supplied scales enters only through learned corrections (Proposition 2, Appendix D). KineTraj meets these three requirements with three components (Figure 2). The scale-invariant kinematic encoder maps each window to the normalized coordinates of Definition 1; the domain-agnostic backbone summarizes the normalized window into a representation $\mathbf{h}$ without any domain input; and the KineMix decoder learns the shared distribution as a mixture over a few motion regimes, with closed-form physics anchors as the regimes, a gate as the regime posterior, and learned residual modes as the within-regime deviation, and rescales its output by s_y.

Mixture form. To give the shared distribution a concrete form, we follow the multiple-model view of target tracking (Blom & Bar-Shalom, 1988; Li & Jilkov, 2003) and describe motion over a short window through *regimes*. A regime is a characteristic way a body can move over a few time steps, such as holding its speed or turning at a steady rate. Writing $A^{(m)}(\mathbf{k}')$ for the normalized future that regime m would produce from the normalized past, $\pi_m(\mathbf{k}')$ for the probability that the window is in regime m, and D_m for the distribution of the gap between the realized future and the regime's future, the shared distribution takes the form of a mixture over regimes,

$$Q(\mathbf{y}' \mid \mathbf{k}') = \sum_m \pi_m(\mathbf{k}')\, D_m\big(\mathbf{y}' - A^{(m)}(\mathbf{k}')\big). \tag{5}$$

Newtonian kinematics fixes only how each regime continues; which regimes occur and how motion scatters around them is what the shared distribution carries.

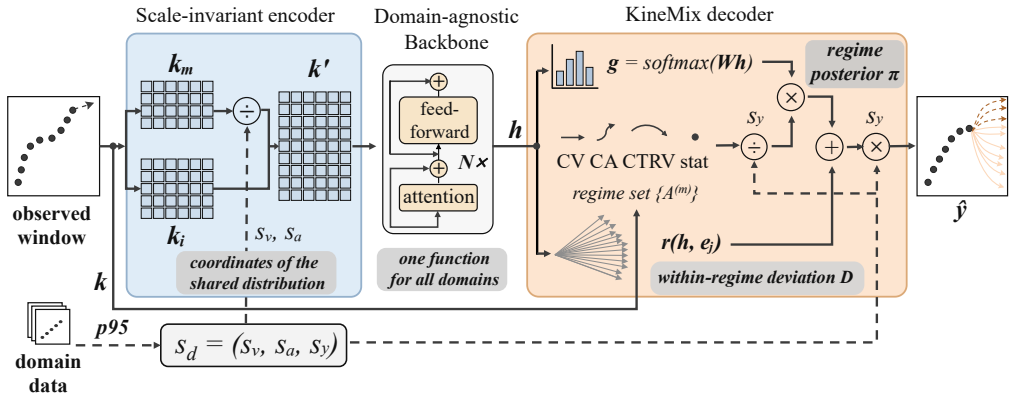


Figure 2: The framework of KINETRAJ. The scale-invariant encoder and the domain-agnostic backbone (Enc) produce the representation $\mathbf{h}$ that enters the gate $\mathrm{softmax}(\mathbf{Wh})$ and the residual $r(\mathbf{h}, \mathbf{e}_j)$. The raw window $\mathbf{k}$, carried by the solid line below the encoder, determines four fixed physics anchors; the gate sets their mixture weights and the residual their learned corrections. Rescaling by s_y returns the sixteen residual-corrected modes to physical units; the four anchors, which the model returns as further hypotheses, are not drawn separately. Dashed lines carry the observed-input scales s_d.

Forward pass. For an observed window $\mathbf{k}$ in physical units, a hypothesis of KINETRAJ is

$$\hat{\mathbf{y}} = s_y \Big[\bar{A}(\mathbf{k})/s_y + \mathbf{r} \Big], \qquad \bar{A}(\mathbf{k}) = \textstyle\sum_m g_m(\mathbf{h})\, A^{(m)}(\mathbf{k}), \qquad \mathbf{h} = \mathrm{Enc}\big(\phi_{s_d}(\mathbf{k})\big), \tag{6}$$

The physics anchors $A^{(m)}$ are the regime set of Eq. (5), the gate weights g_m its posterior π, and the learned residual $\mathbf{r}$, in normalized coordinates, a draw from its deviation D.

4.2 SCALE-INVARIANT ENCODER

Kinematic channels. A window covers T_o observed positions and the target the next T steps. The kinematic description $\mathbf{k}$ stacks per step the speed, acceleration, and turn rate, which form the magnitude channels $\mathbf{k}_\mathrm{m}$, and the normalized position, heading, and local straightness, the ratio of net to travelled distance, which form the scale-invariant channels $\mathbf{k}_\mathrm{i}$ (Appendix A.5). The characteristic scales of a domain are $s_d = (s_v, s_a, s_y)$, and Eq. (1) divides v_t by s_v, a_t by s_a, and ω_t by s_a/s_v, the reciprocal of the domain's characteristic time; the target $\mathbf{y} \in \mathbb{R}^{2T}$ stacks the T future displacements and is normalized by s_y. The raw positions enter only the physics anchors below, through the last observed velocity $\mathbf{v}$, acceleration $\mathbf{a}$, and turn rate ω.

Normalization. The encoder applies ϕ_{s_d} of Eq. (1) with the characteristic scales of the window's domain. All three scales are 95th percentiles of the domain's observed windows, of the speed, of the absolute change of speed per unit time, and of the per-step displacement (Appendix A.2), so that no hand-set value enters the model.

Domain-agnostic backbone. A Transformer summarizes the normalized window into $\mathbf{h} = \mathrm{Enc}(\mathbf{k}')$; it receives no domain label and no absolute scale, so every quantity computed from $\mathbf{h}$ is scale-free by construction, and the same weights serve every domain, the single function that Section 4.1 requires.

4.3 KINEMIX DECODER

Regime set. The regime set $\{A^{(m)}\}$ is realized by $M = 4$ physics anchors, deterministic futures computed in closed form from the last observed step and measured cumulatively from the last observed point. With velocity $\mathbf{v}$ of speed $\|\mathbf{v}\|$ and heading $\theta_0 = \angle\mathbf{v}$, acceleration $\mathbf{a}$, turn rate ω, and steps $t = 1, \ldots, T$:

$$A_t^{\mathrm{CV}} = t\,\mathbf{v}, \qquad A_t^{\mathrm{CA}} = t\,\mathbf{v} + \tfrac{1}{2}t^2\mathbf{a}, \qquad A_t^{\mathrm{stat}} = \mathbf{0},$$
$$A_t^{\mathrm{CTRV}} = \tfrac{\|\mathbf{v}\|}{\omega}\big(\sin(\theta_0 + \omega t) - \sin\theta_0,\ \cos\theta_0 - \cos(\theta_0 + \omega t)\big)^\top.$$

CTRV reduces to CV as $\omega \to 0$. The anchors take the raw window in physical units, contain no learned parameters, and are homogeneous of degree one, $A(s \cdot \mathbf{k}) = s\,A(\mathbf{k})$, so that dividing them by s_y gives the normalized regimes of Eq. (5); what the domains share is that a small homogeneous family suffices, not which members it has.

Regime posterior. The posterior π is a gate over the anchors, $\mathbf{g} = \mathrm{softmax}(\mathbf{W}\mathbf{h}) \in \Delta^M$, which yields the mixed anchor $\bar{A}_t = \sum_m g_m A_t^{(m)}$. Since $\mathbf{h}$ carries no absolute scale, the gate assigns a window to regimes from scale-invariant cues only, such as turn rate, normalized acceleration, and straightness, as the principle requires.

Within-regime deviation. The deviation D of Eq. (5) is realized by a residual head $r_\theta(\mathbf{h}, \mathbf{e}_j)$ driven by K_r learned mode queries $\mathbf{e}_1, \ldots, \mathbf{e}_{K_r}$ in place of Gaussian noise. Each query yields one hypothesis in normalized coordinates, $\hat{\mathbf{y}}_t'^{(j)} = \bar{A}_t/s_y + r_\theta(\mathbf{h}, \mathbf{e}_j)_t$, and multiplying by s_y returns it to physical units; this is the fixed-head mechanism of multimodal predictors (Phan-Minh et al., 2020; Xu et al., 2023). An error in s_y therefore leaves the anchors untouched, since s_y cancels in $\bar{A}/s_y \cdot s_y$, and enters through the residual alone, the $O(|\epsilon|)$ scale term of Section 3.2 (Proposition 2).

Scaling a window and its domain's scales by the same factor s leaves the normalized input unchanged, since θ_0 and ω are scale-free, while the anchors and s_y scale by s, so that KINETRAJ$(s \cdot \mathbf{k};\, s \cdot s_d) = s$ KINETRAJ$(\mathbf{k};\, s_d)$ exactly; absolute magnitudes are divided out at the input and multiplied back at the output, and the learned map never sees them. With the domain-agnostic backbone, KINETRAJ therefore meets the three requirements of Section 4.1, and Proposition 1 applies to it.

5 EXPERIMENTS

5.1 EXPERIMENTAL SETUP

Datasets and protocol. We use a leave-one-domain-out (LODO) protocol (Gulrajani & Lopez-Paz, 2021) over six datasets spanning five mobility domains, ETH/UCY (Pellegrini et al., 2009; Lerner et al., 2007) and SDD (Robicquet et al., 2016) (pedestrian), CT-DMA (Nguyen & Fablet, 2024) (maritime AIS), T-DRIVE (Yuan et al., 2010) (urban taxi GPS), GEOLIFE (Zheng et al., 2009) (multi-modal GPS), and AIRCRAFT (Schäfer et al., 2014) (en-route ADS-B), each a domain with its own scales. For each held-out dataset the model is trained only on the other five, without fine-tuning or adaptation, and at inference receives the observation window and the three 95th-percentile statistics of the held-out domain's observed windows, which contain no future position, no hand-set value, and no domain label; the baselines that normalize per dataset compute their own constant from the same windows. All experiments use the $8 \to 12$ horizon and a fixed test split.

Metrics. Following common practice in multimodal trajectory prediction (Gupta et al., 2018; Salzmann et al., 2020; Mangalam et al., 2020; Yuan et al., 2021; Shi et al., 2022), we report ADE and FDE, in each case the error of the closest of $K = 20$ hypotheses, with the same K on every dataset and every method evaluated with its native hypothesis set.

Baselines. We compare against three groups of methods. The physics references require no training and comprise constant velocity (CV), constant acceleration (CA), the sampled constant-velocity model CVM-S (Schöller et al., 2020), and a best-of-4 physics oracle in the sense of CoverNet (Phan-Minh et al., 2020). The learned predictors are trained on the same sources and comprise the coordinate-only cores of Social-LSTM, AgentFormer, Trajectron++, and Social-GAN (Appendix A.4), UniTraj (Zhu et al., 2025) initialized from its released weights, and the published predictors EqMotion (Xu et al., 2023) and SingularTrajectory (Bae et al., 2024) trained with their own recipes. The foundation models Chronos (Ansari et al., 2024), Time-MoE (Shi et al., 2025), and OmniTraj (Gao et al., 2027) are applied zero-shot with their released weights. CV, CA, UniTraj, and Time-MoE return a single trajectory, and every method receives the same observation windows.

Implementation. The backbone has four Transformer layers with $d_{\mathrm{model}} = 128$; the decoder uses $K_r = 16$ mode queries, which with the four anchors give the $K = 20$ hypotheses; the model has 0.59M parameters and is trained end-to-end with a winner-take-all variety loss (Appendix A).

Table 1: Zero-shot LODO on six datasets. ADE and FDE in metres, nautical miles for CT-DMA; lower is better for every column, best per column in bold. Params in millions of trainable parameters; the physics references have none. Trained models are averaged over random seeds (std. in Table 12).

			ETH/UCY		SDD		CT-DMA		T-Drive		GeoLife		Aircraft	
	Method	Params	ADE	FDE	ADE	FDE	ADE	FDE	ADE	FDE	ADE	FDE	ADE	FDE
Physics	CV	–	0.569	1.234	0.808	1.663	2.872	6.187	854	1901	39.2	81.9	1612	3690
	CA	–	1.246	3.184	3.077	7.763	5.559	13.965	1791	4532	115.1	296.9	3665	9643
	CVM-S	–	0.399	0.812	0.590	1.154	1.411	2.543	697	1481	33.2	66.4	1265	2463
	Physics oracle	–	0.487	1.017	0.721	1.405	2.291	5.009	746	1588	31.6	63.6	1313	2995
Learned	LSTM	0.23	0.895	1.644	1.480	2.591	6.576	11.989	644	1123	62.7	111.8	13055	23739
	Transformer	0.56	0.517	0.912	0.818	1.385	2.923	5.171	662	1167	76.9	137.5	10893	19674
	CVAE	0.31	1.328	2.416	3.141	5.601	10.795	19.522	777	1436	83.6	148.1	15322	28184
	GAN	0.40	0.880	1.541	1.528	2.578	6.675	11.640	658	1138	67.7	111.3	12777	23146
	UniTraj	2.40	0.764	1.471	1.074	2.065	4.604	9.187	903	1794	71.2	133.7	5972	11764
	EqMotion	3.00	0.326	0.632	0.376	0.664	1.875	3.880	464	847	19.5	33.0	1102	2190
	SingularTrajectory	1.90	0.433	0.828	0.413	0.740	1.834	3.412	439	775	18.7	33.0	1905	3385
FM	Time-MoE	50.00	0.641	1.362	0.729	1.520	3.485	7.534	924	1959	39.1	80.3	1926	4371
	Chronos	200.00	0.340	0.655	0.460	0.847	2.287	4.597	593	1186	23.7	44.1	1168	2486
	OmniTraj	7.50	0.284	0.533	**0.342**	0.590	2.138	3.838	418	786	75.5	145.4	3097	5056
Ours	KINETRAJ	0.59	**0.261**	**0.473**	0.349	**0.571**	**1.223**	**2.255**	**368**	**626**	**17.2**	**29.4**	**813**	**1500**

5.2 COMPARISON WITH EXISTING PREDICTORS

Table 1 reports all methods on the six folds, from which three observations follow. First, learned predictors transfer only when they carry an invariance: three of the four coordinate-only cores fall below constant velocity on four of the five in-range folds and all four rise to seven to ten times its error on the aircraft fold, UniTraj remains below constant velocity on every fold, whereas EqMotion improves over constant velocity and Chronos on every fold and SingularTrajectory over constant velocity on every fold but aircraft. Second, Chronos, which standardizes every window before forecasting, improves over constant velocity on every fold without any training on trajectories; KINETRAJ improves over it on all six folds, by 23% to 47% in ADE, with 0.3% of its parameters. Third, the foundation models split by family. OmniTraj, pretrained on human motion, is the strongest baseline on the two pedestrian folds, 2% ahead of KINETRAJ on SDD, but falls behind constant velocity on multi-modal GeoLife and on the aircraft fold, where its error doubles. KINETRAJ improves over every physics reference on every fold and, with EqMotion, is one of the two trained models that never fall below constant velocity; it is the best method on five folds and second to OmniTraj on SDD, with the largest margins on the off-scale aircraft fold (26% over EqMotion) and on CT-DMA (13% over CVM-S), and averaged over the six folds it reduces ADE by 18% and FDE by 22% relative to the stronger of EqMotion and SingularTrajectory on each fold. Holding out both pedestrian sets at once (Appendix B.1) removes the within-family help: trained on the four non-pedestrian sets alone, KINETRAJ still leads on ETH/UCY (ADE 0.274 against 0.284 for OmniTraj and 0.338 for EqMotion) and stays within 2% of OmniTraj on SDD (0.349 against 0.342), its ADE rising by 5% and 0% against 4% for EqMotion on both.

5.3 VERIFYING THE IMPLICATIONS OF THE PRINCIPLE

Coverage in normalized coordinates. Over the six folds the coordinate-only cores and UniTraj, which bind their output magnitude to the training scale, are worse than constant velocity by a factor of 1.5 to 2.9 in the geometric mean, and none of them improves on it on the multi-modal or the off-scale fold, whereas KINETRAJ and the per-window-standardized Chronos improve on it on every fold, as implication (i) states. The gain is one of coverage rather than point accuracy, since the single-trajectory output of KINETRAJ is its constant-velocity anchor and the cross-domain advantage emerges only with a hypothesis set.

Scale extrapolation. The aircraft fold holds out a domain 14 times faster than the fastest source, outside the scale band that the five in-range folds share, and separates methods by mechanism (Figure 3). Methods that regress coordinates in training units rise above constant velocity, the coordinate-only cores to seven to ten times its error, UniTraj to nearly four times, and the pretrained OmniTraj to twice, whereas methods whose output magnitude follows the input scale, CVM-S, the physics oracle, the per-window-standardized Chronos, EqMotion, and KINETRAJ, stay below it; KINETRAJ and EqMotion are the two best methods on the fold, as implication (ii) states for KINETRAJ.

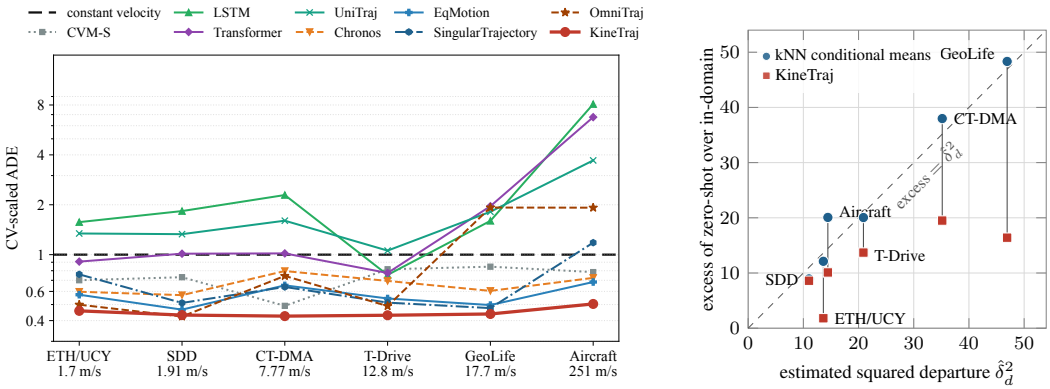


Figure 3: Left: scale extrapolation. Held-out ADE divided by that of constant velocity, folds ordered by s_v. On the aircraft fold, 14 times faster than the fastest source, methods that regress coordinates in training units rise above constant velocity; those whose output magnitude follows the input scale, KINETRAJ among them, stay below it. Right: zero-shot excess over in-domain prediction, in normalized squared error, against the estimated squared departure $\hat{\delta}_d^2$ per fold (Table 4). The kNN conditional means follow Proposition 1; the excess of KINETRAJ ranks with $\hat{\delta}_d^2$ and stays below it.

Departure and transfer. We estimate δ_d per fold with the nearest-neighbour probe of Appendix B.2.1 and measure errors in the proposition's own terms, the normalized squared error of a point prediction, for KINETRAJ the score-weighted mean of its learned modes, trained on the sources and on the held-out set's own training split. For the conditional means themselves the ratio of the measured excess to $\hat{\delta}_d^2$ lies between 0.81 and 1.39 on the six folds, where the identity of Proposition 1 sets it to one, and for KINETRAJ the excess ranks with $\hat{\delta}_d^2$ across the six folds (Figure 3, right; Spearman 0.89, one-sided $p = 0.017$) and stays below $\hat{\delta}_d^2$ on every fold, as implication (iii) states. Proposition 1 fixes the excess of the source-optimal predictor over the Bayes optimum, whereas KINETRAJ is a learned model measured against a learned in-domain model, so its excess is expected only to rank with $\hat{\delta}_d^2$ (Appendix D); averaging all twenty hypotheses weakens the ranking to 0.60, and six domains detect only a strong ordering. Within folds (Appendix B.2.1) the largest departures and excesses fall on the driving windows of GeoLife and the fast windows of T-Drive, as the implication names, and the excess rises with $\hat{\delta}$ across speed tertiles in five of the six folds.

Absolute scale as input. The variant of KINETRAJ whose gate and residual receive $\log u$, the mean observed speed of the window (Appendix B.2.2), changes the in-range error by between -2.9% and $+4.6\%$ across the five folds, gains and losses alike, and is 10% worse on the off-scale aircraft fold, where the absolute speed lies outside every training value and the model that never sees it is unaffected, as implication (iv) states.

Appendix B reports the pedestrian-family fold, the departure probe by speed tertile, the absolute-scale variant, the selection of and sensitivity to the characteristic scales, backbone and hyperparameter robustness, and the seed variability of Table 1.

6 CONCLUSION

We have stated the kinematic similarity principle, that mobility domains share a kinematic distribution of short-horizon motion once expressed in its characteristic scales, up to a departure δ_d per domain. The principle sets the limit of zero-shot prediction and dictates the design of KINETRAJ. On six datasets across five mobility domains, KINETRAJ is the best method on five folds, within 2% of the best foundation model on the sixth, and the benchmark bears out its four implications. The gain is one of multimodal coverage rather than point accuracy, since the single-trajectory output of KINETRAJ is its constant-velocity anchor, and the regime set is fixed and physics-derived; longer horizons, multi-agent interaction, and further domains are natural next steps.

AI USE STATEMENT

In this work, we used generative AI tools as research, programming, and writing assistants. They were used to support brainstorming and the presentation of theoretical ideas and mathematical arguments, provide feedback on proofs and experimental design, assist in implementing the proposed method, write and edit auxiliary code, and support batch experiment execution and experiment management. We also used generative AI tools to assist with document reference formatting.

The experimental protocols, evaluation criteria, and research decisions were determined by the authors. All AI-assisted implementations were reviewed and tested by the authors, and the experimental outputs were checked for correctness and consistency before being included in the paper. All mathematical statements and proofs were independently checked by the authors, and all AI-assisted text, code, figures, and analyses were reviewed and revised by the authors. We take full responsibility for the final content and results of this work.

REFERENCES

Alexandre Alahi, Kratarth Goel, Vignesh Ramanathan, Alexandre Robicquet, Li Fei-Fei, and Silvio Savarese. Social LSTM: Human trajectory prediction in crowded spaces. In *CVPR*, 2016.

R. McNeill Alexander and A. S. Jayes. A dynamic similarity hypothesis for the gaits of quadrupedal mammals. *Journal of Zoology*, 201(1):135–152, 1983.

Abdul Fatir Ansari, Lorenzo Stella, Caner Turkmen, Xiyuan Zhang, Pedro Mercado, Huibin Shen, Oleksandr Shchur, Syama Sundar Rangapuram, Sebastian Pineda Arango, Shubham Kapoor, Jasper Zschiegner, Danielle C. Maddix, Hao Wang, Michael W. Mahoney, Kari Torkkola, Andrew Gordon Wilson, Michael Bohlke-Schneider, and Yuyang Wang. Chronos: Learning the language of time series. *Transactions on Machine Learning Research*, 2024.

Inhwan Bae, Young-Jae Park, and Hae-Gon Jeon. SingularTrajectory: Universal trajectory predictor using diffusion model. In *CVPR*, 2024.

Henk A. P. Blom and Yaakov Bar-Shalom. The interacting multiple model algorithm for systems with Markovian switching coefficients. *IEEE Transactions on Automatic Control*, 33(8):780–783, 1988.

Edgar Buckingham. On physically similar systems; illustrations of the use of dimensional equations. *Physical Review*, 4(4):345–376, 1914.

Ming-Fang Chang, John 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 *CVPR*, 2019.

Henggang Cui, Thi Nguyen, Fang-Chieh Chou, Tsung-Han Lin, Jeff Schneider, David Bradley, and Nemanja Djuric. Deep kinematic models for kinematically feasible vehicle trajectory predictions. In *ICRA*, 2020.

Abhimanyu Das, Weihao Kong, Rajat Sen, and Yichen Zhou. A decoder-only foundation model for time-series forecasting. In *ICML*, 2024.

Yonghao Dong, Le Wang, Sanping Zhou, Gang Hua, and Changyin Sun. Recurrent aligned network for generalized pedestrian trajectory prediction. *IEEE Transactions on Circuits and Systems for Video Technology*, 35(3):2188–2201, 2025.

Scott Ettinger, Shuyang Cheng, Benjamin Caine, Chenxi Liu, Hang Zhao, Sabeek Pradhan, Yuning Chai, Ben Sapp, Charles R. Qi, Yin Zhou, Zoey Yang, Aurélien Chouard, Pei Sun, Jiquan Ngiam, Vijay Vasudevan, Alexander McCauley, Jonathon Shlens, and Dragomir Anguelov. Large scale interactive motion forecasting for autonomous driving: The Waymo open motion dataset. In *ICCV*, 2021.

Lan Feng, Mohammadhossein Bahari, Kaouther Messaoud Ben Amor, Éloi Zablocki, Matthieu Cord, and Alexandre Alahi. UniTraj: A unified framework for scalable vehicle trajectory prediction. In *ECCV*, 2024.

Yang Gao, Po-Chien Luan, Kaouther Messaoud, Lan Feng, and Alexandre Alahi. OmniTraj: Pre-training on heterogeneous data for adaptive and zero-shot human trajectory prediction. *Transportation Research Part C: Emerging Technologies*, 194:105971, 2027.

Francesco Giuliari, Irtiza Hasan, Marco Cristani, and Fabio Galasso. Transformer networks for trajectory forecasting. In *ICPR*, 2020.

Marta C. González, César A. Hidalgo, and Albert-László Barabási. Understanding individual human mobility patterns. *Nature*, 453(7196):779–782, 2008.

Anita Graser. Timeseries foundation models for mobility: A benchmark comparison with traditional and deep learning models. In *CoRR abs/2504.03725*, 2025.

Ishaan Gulrajani and David Lopez-Paz. In search of lost domain generalization. In *ICLR*, 2021.

Agrim Gupta, Justin Johnson, Li Fei-Fei, Silvio Savarese, and Alexandre Alahi. Social GAN: Socially acceptable trajectories with generative adversarial networks. In *CVPR*, 2018.

Pingxuan Huang, Zhenhua Cui, Jing Li, Shenghua Gao, Bo Hu, and Yanyan Fang. Cross-domain trajectory prediction with CTP-Net. In *CICAI*, 2022.

Renhao Huang, Anthony Tompkins, Maurice Pagnucco, and Yang Song. Towards single source domain generalisation in trajectory prediction: A motion prior based approach. In *CoLLAs*, 2023.

Yiheng Huang, Xiaowei Mao, Shengnan Guo, Yubin Chen, Junfeng Shen, Tiankuo Li, Youfang Lin, and Huaiyu Wan. STD-PLM: Understanding both spatial and temporal properties of spatial-temporal data with PLM. In *AAAI*, 2025.

Boris Ivanovic, James Harrison, and Marco Pavone. Expanding the deployment envelope of behavior prediction via adaptive meta-learning. In *ICRA*, 2023.

Alon Lerner, Yiorgos Chrysanthou, and Dani Lischinski. Crowds by example. *Computer Graphics Forum*, 26(3):655–664, 2007.

X. Rong Li and Vesselin P. Jilkov. Survey of maneuvering target tracking. Part I: Dynamic models. *IEEE Transactions on Aerospace and Electronic Systems*, 39(4):1333–1364, 2003.

Xiaohe Li, Feilong Huang, Zide Fan, Fangli Mou, Yingyan Hou, Chen Qian, and Lijie Wen. MetaTra: Meta-learning for generalized trajectory prediction in unseen domain. In *CoRR abs/2402.08221*, 2024.

Zhonghang Li, Long Xia, Lei Shi, Yong Xu, Dawei Yin, and Chao Huang. Open spatio-temporal foundation models for traffic prediction. *ACM Transactions on Intelligent Systems and Technology*, 2025.

Yan Lin, Tonglong Wei, Zeyu Zhou, Haomin Wen, Jilin Hu, Shengnan Guo, Youfang Lin, and Huaiyu Wan. TrajFM: A vehicle trajectory foundation model for region and task transferability. In *CoRR abs/2408.15251*, 2024.

Karttikeya Mangalam, Harshayu Girase, Shreyas Agarwal, Kuan-Hui Lee, Ehsan Adeli, Jitendra Malik, and Adrien Gaidon. It is not the journey but the destination: Endpoint conditioned trajectory prediction. In *ECCV*, 2020.

Duong Nguyen and Ronan Fablet. A transformer network with sparse augmented data representation and cross entropy loss for AIS-based vessel trajectory prediction. *IEEE Access*, 12:21596–21609, 2024.

Stefano Pellegrini, Andreas Ess, Konrad Schindler, and Luc Van Gool. You'll never walk alone: Modeling social behavior for multi-target tracking. In *ICCV*, 2009.

Tung Phan-Minh, Elena Corina Grigore, Freddy A. Boulton, Oscar Beijbom, and Eric M. Wolff. CoverNet: Multimodal behavior prediction using trajectory sets. In *CVPR*, 2020.

Ben A. Rainbow, Qianhui Men, and Hubert P. H. Shum. Semantics-STGCNN: A semantics-guided spatial-temporal graph convolutional network for multi-class trajectory prediction. In *SMC*, 2021.

Alexandre Robicquet, Amir Sadeghian, Alexandre Alahi, and Silvio Savarese. Learning social etiquette: Human trajectory understanding in crowded scenes. In *ECCV*, 2016.

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

Matthias Schäfer, Martin Strohmeier, Vincent Lenders, Ivan Martinovic, and Matthias Wilhelm. Bringing up OpenSky: A large-scale ADS-B sensor network for research. In *IPSN*, 2014.

Christoph Schöller, Vincent Aravantinos, Florian Lay, and Alois Knoll. What the constant velocity model can teach us about pedestrian motion prediction. *IEEE Robotics and Automation Letters*, 5(2):1696–1703, 2020.

Shaoshuai Shi, Li Jiang, Dengxin Dai, and Bernt Schiele. Motion transformer with global intention localization and local movement refinement. In *NeurIPS*, 2022.

Xiaoming Shi, Shiyu Wang, Yuqi Nie, Dianqi Li, Zhou Ye, Qingsong Wen, and Ming Jin. Time-MoE: Billion-scale time series foundation models with mixture of experts. In *ICLR*, 2025.

Tonglong Wei, Yan Lin, Zeyu Zhou, Haomin Wen, Jilin Hu, Shengnan Guo, Youfang Lin, Gao Cong, and Huaiyu Wan. TransferTraj: A vehicle trajectory learning model for region and task transferability. In *NeurIPS*, 2025.

Theodor Westny, David Axelsson, Björn Olofsson, and Erik Frisk. Unveiling transferability in trajectory prediction via latent scene embeddings. In *ECCV*, 2026.

Chenxin Xu, Robby T Tan, Yuhong Tan, Siheng Chen, Yu Guang Wang, Xinchao Wang, and Yanfeng Wang. EqMotion: Equivariant multi-agent motion prediction with invariant interaction reasoning. In *CVPR*, 2023.

Yi Xu, Lichen Wang, Yizhou Wang, and Yun Fu. Adaptive trajectory prediction via transferable GNN. In *CVPR*, 2022.

Javier Yanes-Pulido and Filipe Rodrigues. Time series foundation models as strong baselines in transportation forecasting: A large-scale benchmark analysis. In *CoRR abs/2602.24238*, 2026.

Sean Bin Yang, Ying Sun, Yunyao Cheng, Yan Lin, Kristian Torp, and Jilin Hu. Spatio-temporal trajectory foundation model: Recent advances and future directions. In *CIKM Workshop on STIntelligence*, 2025.

Luyao Ye, Zikang Zhou, and Jianping Wang. Improving the generalizability of trajectory prediction models with Frenet-based domain normalization. In *ICRA*, 2023.

Jing Yuan, Yu Zheng, Chengyang Zhang, Wenlei Xie, Xing Xie, Guangzhong Sun, and Yan Huang. T-Drive: Driving directions based on taxi trajectories. In *SIGSPATIAL*, 2010.

Ye Yuan, Xinshuo Weng, Yanglan Ou, and Kris M. Kitani. AgentFormer: Agent-aware transformers for socio-temporal multi-agent forecasting. In *ICCV*, 2021.

Yuan Yuan, Jingtao Ding, Jie Feng, Depeng Jin, and Yong Li. UniST: A prompt-empowered universal model for urban spatio-temporal prediction. In *KDD*, 2024.

Yu Zheng, Lizhu Zhang, Xing Xie, and Wei-Ying Ma. Mining interesting locations and travel sequences from GPS trajectories. In *WWW*, 2009.

Yuanshao Zhu, James Jianqiao Yu, Xiangyu Zhao, Xun Zhou, Liang Han, Xuetao Wei, and Yuxuan Liang. UniTraj: Learning a universal trajectory foundation model from billion-scale worldwide traces. In *NeurIPS*, 2025.

A EXPERIMENTAL DETAILS

A.1 DATASETS AND PROTOCOL

Datasets. The benchmark comprises six datasets spanning five mobility domains: ETH/UCY (Pellegrini et al., 2009; Lerner et al., 2007) and SDD (Robicquet et al., 2016) (pedestrian), CT-DMA (maritime AIS, the Danish Maritime Authority subset released with TrAISformer (Nguyen & Fablet, 2024)), T-DRIVE (Yuan et al., 2010) (urban taxi GPS), GEOLIFE (Zheng et al., 2009) (multi-modal GPS), and AIRCRAFT (en-route ADS-B collected from the OpenSky Network (Schäfer et al., 2014), whose characteristic speed of 252 m/s is 14 times that of the fastest other domain). Each dataset is one domain of the protocol, with its own scales (Appendix A.2), so the two pedestrian sets form separate folds; when one of them is held out the other remains a source, so these two folds test transfer within the pedestrian family, whereas the other four hold out an entire mobility domain (Appendix B.1 holds out both pedestrian sets at once). The ADS-B snapshot was collected from OpenSky's public interface under its terms of use, which do not permit redistribution; the code release therefore contains the collection script with the exact time window and region rather than the data, and the trained checkpoints. All but GEOLIFE are single-regime sources; GEOLIFE is GPS-logged human mobility that mixes walking, cycling, and driving within one trajectory, so its sub-modes partially overlap the source domains at a different sensing scale, which makes it a compositional, cross-sensing test rather than a wholly new transport mode.

Leave-one-domain-out protocol. We follow the leave-one-domain-out (LODO) protocol (Gulrajani & Lopez-Paz, 2021). For each held-out dataset the model is trained only on the other five, so no trajectory, window, or label from the held-out set is seen at training time, and there is no fine-tuning or adaptation on it. The held-out domain's data source and sampling regime are therefore unseen. The protocol is stricter than few-shot or domain-adaptation transfer, which expose target trajectories, and tests whether a model uses shared structure rather than target examples.

Information about the held-out domain. At inference the model receives the past-observation window and the three scales of the held-out domain (Section 4). Its step-displacement scale is the 95th percentile of the per-step displacement over the observed parts of the held-out evaluation windows, and its speed and acceleration scales are the 95th percentiles of the observed speed and absolute speed change per unit time over the same windows (Appendix A.2), summaries of inputs that contain no future position and no manually specified scale constants. For the source domains the same statistics are taken over their training windows. These three scales are the only domain-level statistics supplied to KINETRAJ, the learned predictor is not conditioned on domain labels. The published trajectory predictors that normalize per dataset compute their own scale constants from the same observed windows, using observed inputs only.

Horizon and splits. All experiments use the standard $8 \rightarrow 12$ horizon, $T_o = 8$ observed steps and $T = 12$ predicted steps, at sampling intervals of 0.4 s (ETH/UCY, SDD), 5 s (GeoLife), 15 s (Aircraft), 60 s (T-Drive), and 600 s (CT-DMA), so the predicted horizon spans 4.8 s to 2 h. Each source domain is capped at the same number of trajectories and windows, 2,000 and 12,000, to balance the training mix, and the held-out test windows form one fixed split.

A.2 CHARACTERISTIC SCALES

Every dataset has its own characteristic scales, a speed scale s_v, an acceleration scale s_a, and a step-displacement scale s_y, and Table 2 lists them per dataset. They are the values by which the scale-invariant encoder normalizes that dataset's windows and by which the decoder rescales its predictions, and they are the only domain-specific quantities in the model. Across the six datasets they differ by more than two orders of magnitude, from $s_v = 1.6$ m/s for ETH/UCY to 252 m/s for Aircraft. The three scales of a dataset are statistics of its own observed windows, computed by one rule shared by all datasets: s_v is the 95th percentile of the observed speed, s_a the 95th percentile of the absolute speed change per unit time, and s_y the 95th percentile of the per-step displacement magnitude, the latter in the raw coordinate unit of the dataset (metres for the pedestrian sets, degrees of latitude and longitude for the GPS, AIS, and ADS-B sets). When a dataset is a source its scales are computed over its training windows, and when it is held out over the observed parts of its evaluation windows, which is why the two columns of Table 2 differ slightly; in both cases only observed positions enter, never a future position. The percentile is a hyperparameter shared by all

six datasets; it was chosen by source-domain validation between the 90th and the 95th percentile (Appendix B.3.1), and no held-out result entered the choice.

Table 2: Characteristic scales per dataset. Each is the 95th percentile over a dataset's observed windows of the speed, of the absolute speed change per unit time, and of the per-step displacement, taken over the training windows when the dataset is a source and over the observed parts of the evaluation windows when it is held out. The step scale is in the dataset's raw coordinate unit.

	s_v (m/s)		s_a (m/s^2)		s_y	
Dataset	source	held out	source	held out	source	held out
ETH/UCY (pedestrian)	1.6	1.7	0.53	0.54	0.638 m	0.680 m
SDD (pedestrian)	1.82	1.91	1.2	1.2	0.726 m	0.759 m
CT-DMA (maritime AIS)	7.92	7.77	0.00094	0.0012	0.0643°	0.0662°
T-Drive (vehicle GPS)	12.6	12.8	0.062	0.062	0.0079°	0.0079°
GeoLife (multi-modal GPS)	18.0	17.7	0.45	0.49	0.00094°	0.00092°
Aircraft (OpenSky ADS-B)	252	251	0.81	0.8	0.0491°	0.0499°

A.3 METRICS

Following common practice in multimodal trajectory prediction (Gupta et al., 2018; Salzmann et al., 2020; Mangalam et al., 2020; Yuan et al., 2021; Shi et al., 2022), we report ADE and FDE, computed from the closest of $K = 20$ hypotheses; lower is better. We use $K = 20$ on every dataset so that the folds are comparable; note that published conventions differ between datasets. The protocol is dictated by the task. Trajectory futures are multimodal, so a single MSE-optimal prediction averages distinct modes into an implausible mean, whereas best-of-K rewards covering the realized mode. We apply it identically to every method that produces multiple hypotheses, each with its native hypothesis set (twenty samples or heads; four kinematic models for the physics oracle). Constant velocity, constant acceleration, UniTraj, and Time-MoE return a single trajectory, so their entries are the same at any K. Errors are in metres for ETH/UCY, SDD, T-Drive, GeoLife, and Aircraft and in nautical miles for CT-DMA, computed by Haversine distance for geographic domains and Euclidean distance for ETH/UCY and SDD.

A.4 BASELINES

We compare against three groups of methods.

Physics references. These require no training: constant velocity (CV), which extrapolates the observed velocity and is a standard baseline in pedestrian and driving benchmarks (Gupta et al., 2018; Salzmann et al., 2020; Chang et al., 2019; Ettinger et al., 2021); constant acceleration (CA); the sampled constant-velocity model CVM-S of Schöller et al. (2020), twenty CV hypotheses with heading perturbations; and a physics oracle in the sense of CoverNet (Phan-Minh et al., 2020), the per-window best of four kinematic models (CV, CA, constant turn rate, stationary).

Coordinate-only cores. The learned predictors are trained on the same LODO sources. Five are coordinate-only models: (a) an LSTM encoder–decoder, the single-agent core of Social-LSTM (Alahi et al., 2016); (b) a Transformer encoder in the style of Trajectory Transformer (Giuliari et al., 2020), the coordinate-only core of AgentFormer (Yuan et al., 2021); (c) a CVAE predictor, the multimodal mechanism of Trajectron++ (Salzmann et al., 2020); and (d) a GAN predictor, the mechanism of Social-GAN (Gupta et al., 2018); each of these four draws twenty samples; and (e) UniTraj (Zhu et al., 2025), the masked-trajectory Transformer pretrained on worldwide GPS traces, initialized from its released weights and trained on the sources with the future steps masked, which returns one trajectory. Social and scene-graph methods are not directly applicable across mobility domains (Section 2) and enter only through these coordinate-only cores.

Published predictors. The learned group also contains two published trajectory predictors run as released and trained on the LODO sources with their own recipes: EqMotion (Xu et al., 2023), whose Euclidean-equivariant layers make its prediction invariant to rotation and translation of the input and which outputs twenty fixed hypotheses, and SingularTrajectory (Bae et al., 2024), which normalizes every trajectory by its own origin, heading, and scale, projects it onto a low-dimensional

singular space learned from the training set, and samples futures with a diffusion model. Neither is designed for cross-domain transfer, but each carries an invariance that is relevant to it. Both are run as single-agent models without a scene map, one trajectory per scene, so their interaction and scene modules are inactive and they receive the same input as every other method, consistent with entering the other social methods through their coordinate-only cores.

Foundation models. These are applied zero-shot with their released weights: the time-series models Chronos (Ansari et al., 2024) and Time-MoE (Shi et al., 2025), which forecast each coordinate as a univariate series of step displacements under their native per-window standardization; and OmniTraj (Gao et al., 2027), a Transformer pretrained on heterogeneous human-motion corpora (pedestrians from driving datasets, sports, and pose datasets) with frame-rate conditioning, so that it transfers across sampling rates within human motion.

A.5 Implementation Details

Kinematic channels. The kinematic description $\mathbf{k} \in \mathbb{R}^{7T_o}$ stacks seven channels per observed step, the position $\tilde{\mathbf{p}}_t \in \mathbb{R}^2$ relative to the first point of the window, divided by the window's extent; the speed v_t, acceleration a_t, heading θ_t, and turn rate ω_t; and the local straightness σ_t, the ratio of net to travelled distance over the preceding five steps. Speed, acceleration, and turn rate form the magnitude channels; the normalized position, heading, and straightness form the scale-invariant channels. The heading enters the backbone as the pair $(\cos\theta_t, \sin\theta_t)$, so the network input has eight values per step. The anchors use the last observed velocity $\mathbf{v}$, with speed v_{T_o} and heading θ_{T_o}, and the last-step acceleration $\mathbf{a}$ and turn rate ω.

Model and training. The backbone has four Transformer layers with $d_{\text{model}} = 128$, four heads, and a feed-forward width of 256; the decoder uses $K_r = 16$ mode queries of dimension 16, which with the four anchors give the $K = 20$ hypotheses of the protocol; the model has 0.59M parameters. KineTraj is trained end-to-end with a winner-take-all variety loss over its residual-mode hypotheses. We use the AdamW optimizer with learning rate 5×10^{-4}, weight decay 10^{-5}, and a cosine schedule, a batch size of 64, and at most 50 epochs with early stopping on the source-domain validation error, each domain's error normalized by its initial value so that no domain dominates through its physical scale. Each source domain is capped at 2,000 trajectories and 12,000 training windows, and the data split is fixed across seeds.

B Additional Experiments

This appendix collects the experiments that supplement Section 5. Appendix B.1 adds a stricter pedestrian fold, Appendix B.2 reports the two experiments behind implications (iii) and (iv), Appendix B.3 contains the ablation and sensitivity analyses referred to in Section 5, and Appendix B.4 reports the seed variability of the main results. All errors use ADE unless stated otherwise, and entries of trained models are the mean ± standard deviation over random seeds (Appendix B.4).

B.1 Pedestrian-Family Fold

The ETH/UCY and SDD folds of Table 1 keep the other pedestrian dataset among the sources. To remove this within-family help we add a stricter fold in which *both* pedestrian sets are held out. Every trained model is trained on CT-DMA, T-Drive, GeoLife, and Aircraft only and evaluated zero-shot on ETH/UCY and on SDD (Table 3). The zero-shot methods repeat their Table 1 entries.

B.2 Verifying the Implications of the Principle

B.2.1 Departure Probe

Nearest-neighbour estimator. The departure δ_d of Definition 2 is estimated without training. For each test window of the held-out fold we take its $N = 10$ nearest neighbours in normalized input space, once among the held-out dataset's own training trajectories, windowed densely (stride 5, up to 20 windows per trajectory, at most 200 000 windows; for SDD, whose tracks are single windows, every SDD track except the test windows, 11 087 in all), and once among the pooled source train-

Table 3: Pedestrian-family fold. Every trained model sees only CT-DMA, T-Drive, GeoLife, and Aircraft and is evaluated zero-shot on both pedestrian sets. The physics references and the zero-shot foundation models are deterministic and repeat their Table 1 entries. Errors are ADE and FDE in metres, and the best entry in each column is set in bold.

		ETH/UCY		SDD	
	Method	ADE	FDE	ADE	FDE
Physics	CV	0.569	1.234	0.808	1.663
	CA	1.246	3.184	3.077	7.763
	CVM-S	0.399	0.812	0.590	1.154
	Physics oracle	0.487	1.017	0.721	1.405
Learned	LSTM	1.010±0.043	1.820±0.073	1.850±0.013	3.243±0.008
	Transformer	0.779±0.096	1.363±0.186	1.602±0.095	2.681±0.157
	CVAE	1.837±0.165	3.298±0.290	3.665±0.335	6.416±0.592
	GAN	1.228±0.130	2.104±0.157	1.882±0.176	3.204±0.225
	UniTraj	0.796±0.027	1.550±0.115	0.930±0.074	1.869±0.210
	EqMotion	0.338±0.021	0.657±0.034	0.392±0.001	0.698±0.007
	SingularTrajectory	0.425±0.013	0.809±0.026	0.418±0.011	0.760±0.032
FM	Time-MoE	0.641	1.362	0.729	1.520
	Chronos	0.340	0.655	0.460	0.847
	OmniTraj	0.284	0.533	**0.342**	0.590
Ours	**KINETRAJ**	**0.274**±0.003	**0.515**±0.008	0.349±0.010	**0.560**±0.029

ing windows of the fold, and average their normalized futures; the two averages estimate $g_d^\star(\mathbf{k}')$ and $g^\star(\mathbf{k}')$, and $\hat{\delta}_d$ is the root-mean-square of their gap over the test windows, minus the finite-N variance of the two neighbourhood means. The neighbour count N is unrelated to the K of the best-of-K metric. Inputs are normalized exactly as for KINETRAJ; every input dimension is divided by its standard deviation over the source pool, dimensions that are constant over the pool are dropped, and standardized values are clipped at ten standard deviations, without which one dimension dominates the distance.

Model-free check. The same two conditional-mean estimates check Proposition 1 without a trained model, since the squared risk of the source estimate on the test windows minus that of the in-domain estimate should equal $\hat{\delta}_d^2$. Table 4 reports both together with the squared error of KINETRAJ's point prediction, zero-shot and trained in domain, and Table 5 repeats the comparison within each fold by speed tertile.

Size of the in-domain pool. The pool matters. With the sparse windowing used for training (1 600 to 12 000 windows per set) the in-domain neighbourhoods are wider than the source ones and the in-domain conditional mean is over-smoothed, which lowers the estimate on GeoLife (5.3 against 7.4 with the dense pool at $N = 20$, because sparse neighbourhoods average over walking and driving windows) and inflates it on SDD (4.3 against 3.3). With the dense pools the nearest in-domain neighbours are closer than the nearest source neighbours on every fold.

Enlarging the ETH/UCY evaluation. Under our evaluation protocol, the ETH/UCY test split contains only 175 windows. We therefore repeated the speed-tertile comparison across twelve random partitions of the trajectories into training, validation and test sets. For each split an in-domain model was trained on its training trajectories and evaluated on its test windows, the departure was estimated with that split's training trajectories as the in-domain pool, and the per-window quantities were pooled over splits (1 235 distinct windows). The tertile excesses are 1.18 $[0.46, 1.94]$, 1.87 $[0.97, 2.83]$, and 1.33 $[-0.09, 2.65]$ (bootstrap 95% intervals), so the ordering within ETH/UCY is not resolvable at this sample size either.

Table 4: Departure and zero-shot excess per held-out fold, on the test windows in normalized squared error. The departure $\hat{\delta}_d$ is the kNN estimate at $N = 10$ neighbours, with its range over $N = 5$ to 20 in parentheses, where N is the neighbour count and not the K of the best-of-K metric. Under the conditional means, the squared risk of the pooled-source conditional mean minus that of the in-domain conditional mean, both kNN estimates on the same windows, is divided by $\hat{\delta}_d^2$, a ratio that Proposition 1 predicts to be one. The KINETRAJ columns give the squared error of the score-weighted mean of its learned modes when trained on the sources and when the same model is trained on the held-out set's own training split, together with their difference. Across the six folds that excess ranks with $\hat{\delta}_d^2$, at Spearman 0.89 with a one-sided exact $p = 0.017$. The estimator is described in full at the start of this section.

		Conditional means	KINETRAJ		
Held-out fold	$\hat{\delta}_d$ (N=5–20)	excess/$\hat{\delta}_d^2$	zero-shot	in-domain	excess
ETH/UCY	3.69 (3.57–3.91)	0.89	19.1	17.3	1.8
SDD	3.32 (3.28–3.39)	0.81	54.8	46.2	8.6
CT-DMA	5.93 (5.80–6.02)	1.08	45.6	26.1	19.5
T-Drive	4.57 (4.50–4.60)	0.96	57.1	43.4	13.7
GeoLife	6.85 (6.65–7.36)	1.03	33.6	17.2	16.4
Aircraft	3.80 (3.80–3.83)	1.39	19.5	9.4	10.1

Table 5: Departure and zero-shot excess by speed tertile within each held-out fold, n windows per tertile. $\hat{\delta}$ is the departure estimate, ZS and ID the normalized squared errors of KINETRAJ trained on the sources and in domain, and excess their difference, as in Table 4.

		Slow tertile			Mid tertile			Fast tertile		
Held-out fold	n	$\hat{\delta}$	ZS/ID	excess	$\hat{\delta}$	ZS/ID	excess	$\hat{\delta}$	ZS/ID	excess
ETH/UCY	58	2.49	16.7/13.4	3.3	3.15	20.9/21.4	−0.4	4.96	19.8/17.2	2.6
SDD	67	1.22	11.0/11.8	−0.9	3.87	34.8/28.9	6.0	4.09	118.4/97.7	20.7
CT-DMA	133	3.06	18.6/16.1	2.5	5.61	53.1/31.9	21.2	8.02	64.7/30.2	34.5
T-Drive	1366	0.52	2.3/2.3	0.0	2.40	40.2/37.3	2.9	7.53	128.8/90.6	38.3
GeoLife	4445	0.16	9.7/9.7	0.0	0.71	4.0/3.8	0.1	11.84	87.1/38.2	48.9
Aircraft	91	2.93	28.9/25.2	3.7	3.83	9.8/2.0	7.8	4.47	19.7/1.1	18.6

B.2.2 ABSOLUTE SCALE AS INPUT

KINETRAJ never sees the absolute scale: gate and residual read $\mathbf{h} = \mathrm{Enc}(\mathbf{k}')$, and the anchors are divided and multiplied by the same s_y. Within the training range the absolute speed u of a window reveals its domain, so a variant that appends $\log u$ to the input of the gate and the residual can recover each source domain's own rule and reduce its departure δ_d. Table 6 reports it under the LODO protocol together with the share of the normalized prediction carried by the residual, $\|r_\theta\|/\|\bar{A}/s_y\|$ averaged over held-out windows and mode queries. The variant changes the in-range error by -0.1% (ETH/UCY), -2.9% (SDD), $+4.6\%$ (CT-DMA), $+0.2\%$ (T-Drive), and 0.0% (GeoLife), and raises the aircraft error by 10% (813 to 895 m). In range $\log u$ identifies the source domain of a window, so the variant can learn each source's own rule; the changes are small and of either sign, as the bound by the source departures allows. On aircraft $\log u$ lies outside every training value, the gate and residual extrapolate along that input, and the anchors alone keep the loss to ten percent. This is the source departures seen from the model side, the gain being what not seeing the scale forgoes in range and the loss what it buys beyond it. The residual share, whose median over the six folds Table 6 reports, is 0.51–0.68 on ETH/UCY, SDD, CT-DMA, and Aircraft, where the gated anchors carry most of the prediction; it exceeds one on T-Drive and GeoLife, whose held-out windows include many near-stationary segments for which the anchor mix is close to zero and the residual modes supply the whole hypothesis set.

Table 6: Absolute scale as input. The variant appends $\log u$, the window's mean observed speed in metres per second, to the gate and residual inputs and leaves the anchors unchanged. The residual share is the median over the six folds of $\overline{\|r_\theta\|}/\overline{\|\bar{A}/s_y\|}$, the mean residual norm divided by the mean norm of the gated anchor mix. ADE in metres, nautical miles for CT-DMA.

	ETH/UCY	SDD	CT-DMA	T-Drive	GeoLife	Aircraft	Residual share
KINETRAJ	0.261±0.003	0.349±0.007	1.223±0.014	368±3	17.2±0.1	813±11	0.62±0.028
KINETRAJ + $\log u$ input	0.260±0.003	0.339±0.003	1.279±0.053	369±2	17.2±0.1	895±19	0.72±0.034

B.3 ABLATIONS AND SENSITIVITY

B.3.1 CHOICE AND SENSITIVITY OF THE CHARACTERISTIC SCALES

Which scales are statistics, and which percentile. Table 7 compares the adopted rule, all three scales as 95th-percentile statistics of the observed windows, with the alternatives we trained under the same protocol: the 90th percentile, a partial rule in which only the step scale s_y is a statistic while s_v and s_a stay hand-set constants, and the fully hand-set table used in an earlier version of this work. Two criteria are reported. The source-domain validation error is the target-independent criterion. For each fold it is the mean over the five source domains of the validation ADE at the selected epoch, divided by the constant-velocity error of that domain, averaged over the six folds; it never touches the held-out domain. The held-out ADE is the quantity of Table 1. Within the all-statistics design, source validation prefers the 95th percentile to the 90th (0.385 against 0.388), and that is the only choice made by this criterion. The partial rule has a lower source validation error still (0.377–0.378), so the all-statistics design is not selected by validation; it is adopted because it removes every hand-set value from the model and makes the equivariance of Section 4 exact for all channels, and its held-out results are reported alongside the alternatives so that the reader can weigh the two.

Table 7: Scale rules trained under the same protocol. A statistic is the given percentile of the observed windows, whereas a constant is taken from the hand-set table of an earlier version. Source validation is the target-independent criterion and never touches the held-out domain. The first row is the model of Table 1, and ADE is in metres, nautical miles for CT-DMA.

s_v, s_a	s_y	source val.	ETH/UCY	SDD	CT-DMA	T-Drive	GeoLife	Aircraft
statistics (p95)	p95	0.385±0.001	0.261±0.003	0.349±0.007	1.223±0.014	**368**±3	**17.2**±0.1	**813**±11
statistics (p90)	p90	0.388±0.001	0.265±0.005	**0.339**±0.006	**1.192**±0.022	371±6	17.3±0.1	843±28
constants	p95	0.378±0.001	0.260±0.003	0.353±0.003	1.208±0.013	371±2	20.7±0.3	884±14
constants	p90	**0.377**±0.001	**0.249**±0.003	0.346±0.012	1.245±0.059	370±2	19.0±0.5	894±29
constants	constant	0.388±0.002	0.293±0.004	0.361±0.010	1.509±0.085	398±5	20.5±0.4	1052±22

Sensitivity to the held-out step scale. Table 8 varies only KINETRAJ's held-out step scale s_y at test time, keeping each checkpoint and the other scales fixed. Changing s_y affects only the residual modes, leaving the anchors and gate unchanged (Proposition 2). On ETH/UCY, SDD, CT-DMA, and Aircraft, ADE stays within 9% of its minimum among the tabulated settings from the 75th to the 99th percentile. On T-Drive and GeoLife, ADE falls steeply from the median to the 90th percentile. At the 95th percentile, both are within 3% of their respective minima in the sweep.

B.3.2 BACKBONE ROBUSTNESS

Table 9 replaces the Transformer backbone while keeping the physics-mixture decoder, with its four anchors and sixteen learned modes, and the characteristic scales fixed. Across the six folds the four backbones are nearly indistinguishable. No backbone is uniformly best: the Transformer has the lowest mean error on ETH/UCY and Aircraft, the GRU on GeoLife, and the MLP on CT-DMA, two backbones tie on SDD and on T-Drive, and on every fold the four backbones lie within 5% of one another. This supports the interpretation that the transferable structure is supplied by the scale-normalized physics decoder and its input scales, not by a particular sequence backbone.

Table 8: Held-out step-scale sensitivity of KINETRAJ across six datasets. Checkpoints were trained with the 95th-percentile scale rule. At test time, s_y is set to the given percentile of observed per-step displacement, with s_v, s_a, and model weights fixed. ADE is in metres, nautical miles for CT-DMA; the best tabulated value per dataset is bold.

Percentile	ETH/UCY	SDD	CT-DMA	T-Drive	GeoLife	Aircraft
50th (median)	0.272±0.002	0.360±0.002	1.350±0.047	575±2	25.5±0.0	809±8
75th	0.260±0.003	**0.346**±0.002	1.269±0.043	465±4	22.8±0.1	**799**±9
90th	**0.254**±0.002	0.347±0.005	1.227±0.029	394±3	18.0±0.1	806±11
95th	0.261±0.003	0.349±0.007	**1.223**±0.014	368±3	**17.2**±0.1	813±11
99th	0.276±0.002	0.363±0.006	1.234±0.013	**360**±2	19.5±0.1	834±15

Table 9: Backbone ablation on the six LODO folds. The physics-mixture decoder with its sixteen learned modes is fixed while the backbone is replaced, and the Transformer row is the model of Table 1. ADE in metres, nautical miles for CT-DMA.

	ETH/UCY	SDD	CT-DMA	T-Drive	GeoLife	Aircraft
Transformer (default)	0.261±0.003	0.349±0.007	1.223±0.014	368±3	17.2±0.1	813±11
LSTM	0.268±0.007	0.337±0.004	1.230±0.032	370±2	17.2±0.0	823±3
GRU	0.272±0.005	0.337±0.001	1.246±0.071	369±4	17.1±0.1	840±17
MLP	0.266±0.006	0.342±0.007	1.190±0.024	368±2	17.3±0.1	833±24

B.3.3 HYPERPARAMETER SENSITIVITY

We vary width, depth, and the dimension of the learned mode queries one at a time around the default configuration (Table 10). The default is within 2.3% of the best setting on five folds and about 4% on SDD, where a deeper backbone ($n_{\text{layers}} = 6$) is best (0.335 against 0.349 m); a wider backbone ($d_{\text{model}} = 256$) is best on CT-DMA (1.195 against 1.223 nautical miles), and a wider mode query is best on Aircraft (803 against 813 m), each at a small cost elsewhere and mostly within one to two standard deviations. The improvement is therefore not isolated to one configuration, and the remaining headroom is a few percent of capacity tuning rather than a property of the decoder design.

Table 10: Hyperparameter sensitivity on the six LODO folds. Width, depth, and the dimension of the learned mode queries are varied one at a time around the default of Table 1, which has $d_{\text{model}} = 128$, four layers, and mode dimension 16, with ADE in metres and nautical miles for CT-DMA.

	ETH/UCY	SDD	CT-DMA	T-Drive	GeoLife	Aircraft
default	0.261±0.003	0.349±0.007	1.223±0.014	368±3	17.2±0.1	813±11
$d_{\text{model}} = 64$	0.260±0.009	0.338±0.009	1.200±0.040	367±3	17.2±0.0	828±18
$d_{\text{model}} = 256$	0.268±0.002	0.336±0.003	1.195±0.029	362±1	17.2±0.1	819±24
$n_{\text{layers}} = 2$	0.264±0.004	0.337±0.005	1.220±0.025	368±3	17.2±0.1	833±13
$n_{\text{layers}} = 6$	0.264±0.007	0.335±0.016	1.204±0.049	365±1	17.3±0.0	817±17
mode dim $= 8$	0.262±0.002	0.337±0.010	1.239±0.026	366±2	17.1±0.1	828±11
mode dim $= 32$	0.263±0.004	0.345±0.006	1.212±0.040	368±3	17.2±0.1	803±17

B.3.4 PARAMETER EFFICIENCY

KINETRAJ has 0.59M parameters: one four-layer backbone, two small heads, and sixteen mode queries. Its closed-form physics anchors add no trainable parameters. Table 11 compares model size with the geometric mean of ADE/CV over the six LODO folds, each method with its native K. KINETRAJ is comparable in size to the coordinate-only learned baselines and is 13 to 340 times smaller than the foundation models. This analysis is descriptive rather than a top-1 accuracy claim; the ratio measures best-of-K multimodal coverage.

Table 11: Parameter efficiency and best-of-K coverage. The ratio is the geometric mean over the six folds of ADE divided by the constant-velocity error, each method at its native K and single-trajectory methods at $K = 1$, so that a value below one improves over constant velocity. The best ratio is set in bold.

Method	#Parameters	ADE/CV ↓
LSTM	0.23M	2.00
CVAE	0.31M	2.93
GAN	0.40M	2.04
Transformer	0.56M	1.46
UniTraj	2.40M	1.65
KINETRAJ	0.59M	**0.45**
Time-MoE	50.00M	1.08
Chronos	200.00M	0.66
OmniTraj	7.50M	0.81
EqMotion	3.00M	0.56
SingularTrajectory	1.90M	0.64

B.4 SEED VARIABILITY OF THE MAIN RESULTS

Every trained model of Table 1 was trained three times with different initialisation seeds on the same data split, each under its own configuration, and Table 1 reports the mean. Table 12 adds the standard deviations. The ADE of KINETRAJ varies by 0.5% to 2.0% of its mean across the six folds, against up to 10% for the coordinate-only cores, 6% for EqMotion and SingularTrajectory, and 23% for UniTraj on the aircraft fold. Its lead on five folds exceeds this variation; on SDD its gap to OmniTraj, 0.007 m, equals one standard deviation. The other appendix tables report the same three seeds as mean and standard deviation.

C EXTENDED RELATED WORK

Pedestrian domain generalization. Another line studies domain adaptation or domain generalization for pedestrian forecasting. CTP-Net (Huang et al., 2022) aligns source and target pedestrian feature spaces for cross-camera/domain adaptation; RAN (Dong et al., 2025) removes the need for target-domain pedestrian trajectories by aligning recurrent pedestrian representations; MetaTra (Li et al., 2024) uses meta-learning to simulate source-to-target generalization across unseen trajectory domains. These works are genuine zero-shot or adaptation studies, but their held-out domains are new pedestrian scenes. The agent family, social-interaction assumptions, sampling conventions, and typical motion scale remain comparable. Our setting removes that support: the held-out domain may be maritime AIS, pedestrian motion, vehicle GPS, or multi-modal human GPS, so transfer must be based on a more primitive representation of motion rather than scene-specific pedestrian priors.

Spatio-temporal foundation models. A related family of *spatio-temporal* foundation models—UniST (Yuan et al., 2024), STD-PLM (Huang et al., 2025), and OpenCity (Li et al., 2025)—attains impressive zero-shot generalization *across cities*, but for *aggregate traffic*: they forecast flow or demand over a spatial grid (UniST) or over the nodes of a road graph (STD-PLM, OpenCity). Their reliance on a shared spatial support makes them hard to apply directly to trajectory transfer across mobility domains, where no such support exists and each agent is a moving point, which underscores that current spatio-temporal foundation models target a different problem than cross-domain trajectory transfer. In short, *neither* foundation-model family solves our setting, and for opposite reasons. General time-series models are readily applicable but lack a representation of scale-invariant motion structure, so that Chronos improves over constant velocity on every fold yet trails a model in scale-normalized coordinates with 0.3% of its parameters on every fold (Section 5), whereas spatio-temporal traffic models capture rich spatial structure by assuming a shared graph or grid.

Table 12: Table 1 with the variation over random seeds. Each trained model shows the mean ± standard deviation on the fixed data split; the physics references and the zero-shot foundation models are deterministic and appear once. Units as in Table 1, best mean per column in bold; the six datasets are split over two panels.

		ETH/UCY		SDD		CT-DMA	
	Method	ADE	FDE	ADE	FDE	ADE	FDE
Physics	CV	0.569	1.234	0.808	1.663	2.872	6.187
	CA	1.246	3.184	3.077	7.763	5.559	13.965
	CVM-S	0.399	0.812	0.590	1.154	1.411	2.543
	Physics oracle	0.487	1.017	0.721	1.405	2.291	5.009
Learned	LSTM	0.895±0.063	1.644±0.137	1.480±0.006	2.591±0.016	6.576±0.506	11.989±0.939
	Transformer	0.517±0.050	0.912±0.079	0.818±0.021	1.385±0.046	2.923±0.168	5.171±0.374
	CVAE	1.328±0.052	2.416±0.081	3.141±0.233	5.601±0.447	10.795±0.980	19.522±1.794
	GAN	0.880±0.004	1.541±0.044	1.528±0.047	2.578±0.086	6.675±0.594	11.640±0.960
	UniTraj	0.764±0.069	1.471±0.126	1.074±0.064	2.065±0.139	4.604±0.683	9.187±0.891
	EqMotion	0.326±0.014	0.632±0.020	0.376±0.008	0.664±0.042	1.875±0.091	3.880±0.155
	SingularTrajectory	0.433±0.012	0.828±0.028	0.413±0.007	0.740±0.022	1.834±0.051	3.412±0.014
FM	Time-MoE	0.641	1.362	0.729	1.520	3.485	7.534
	Chronos	0.340	0.655	0.460	0.847	2.287	4.597
	OmniTraj	0.284	0.533	**0.342**	0.590	2.138	3.838
Ours	**KINETRAJ**	**0.261**±0.003	**0.473**±0.017	0.349±0.007	**0.571**±0.018	**1.223**±0.014	**2.255**±0.012

		T-Drive		GeoLife		Aircraft	
	Method	ADE	FDE	ADE	FDE	ADE	FDE
Physics	CV	854	1901	39.2	81.9	1612	3690
	CA	1791	4532	115.1	296.9	3665	9643
	CVM-S	697	1481	33.2	66.4	1265	2463
	Physics oracle	746	1588	31.6	63.6	1313	2995
Learned	LSTM	644±13	1123±22	62.7±1.3	111.8±2.1	13055±62	23739±120
	Transformer	662±36	1167±68	76.9±6.8	137.5±13.1	10893±112	19674±191
	CVAE	777±35	1436±71	83.6±3.7	148.1±6.0	15322±34	28184±57
	GAN	658±8	1138±20	67.7±2.2	111.3±5.0	12777±64	23146±79
	UniTraj	903±24	1794±22	71.2±2.9	133.7±7.8	5972±1378	11764±1735
	EqMotion	464±4	847±13	19.5±0.5	33.0±0.7	1102±70	2190±66
	SingularTrajectory	439±4	775±6	18.7±0.0	33.0±0.3	1905±101	3385±176
FM	Time-MoE	924	1959	39.1	80.3	1926	4371
	Chronos	593	1186	23.7	44.1	1168	2486
	OmniTraj	418	786	75.5	145.4	3097	5056
Ours	**KINETRAJ**	**368**±3	**626**±8	**17.2**±0.1	**29.4**±0.2	**813**±11	**1500**±85

D PROOFS

Proposition 1 gives the excess risk of the ideal zero-shot predictor, the one that applies the exact conditional mean $g^\star$ of the pooled sources with exact scales. This appendix first proves that identity and then follows the three ways in which a real predictor departs from the ideal one. Its map is learned from finite data, its scales are estimated from the observed windows, and it is scored by the bounded best-of-K error rather than by squared loss. Between the first two steps the identity is applied to the source domains, the origin of implication (iv) of Section 3.3.

Setting. Throughout, characteristic scales are positive, every domain has finite second moments, $\mathbb{E}_{P_d}\|\mathbf{y}\|^2 < \infty$, and $g^\star$ is defined on the normalized inputs of the domain under consideration, which holds when the sources cover those inputs in normalized coordinates; otherwise $g^\star$ is fixed by a measurable extension and the departure is computed with the same extension.

The identity. The proof rests on a single fact. Within a domain the normalization is invertible, so conditioning on the raw window and on the normalized window is the same thing, and the in-domain Bayes predictor is the domain's own conditional mean $g_d^\star$ returned to physical units.

Proof of Proposition 1. Within domain $d^\star$ the normalization $\phi_{s_{d^\star}}$ rescales the magnitude channels by positive constants and leaves the other channels unchanged, so it is invertible and the σ-algebras generated by $\mathbf{k}$ and by $\mathbf{k}' = \phi_{s_{d^\star}}(\mathbf{k})$ coincide. Hence $\mathbb{E}_{P_{d^\star}}[\mathbf{y} \mid \mathbf{k}] = \mathbb{E}_{P_{d^\star}}[\mathbf{y} \mid \mathbf{k}'] = s_y\, \mathbb{E}_{Q_{d^\star}}[\mathbf{y}' \mid \mathbf{k}'] = s_y\, g^\star_{d^\star}(\mathbf{k}')$ by Definition 1. Subtracting this from $\hat{\mathbf{y}}(\mathbf{k}) = s_y\, g^\star(\mathbf{k}')$ gives $\hat{\mathbf{y}}(\mathbf{k}) - \mathbb{E}[\mathbf{y} \mid \mathbf{k}] = s_y\big(g^\star(\mathbf{k}') - g^\star_{d^\star}(\mathbf{k}')\big)$, and the expected squared norm over $\mathbf{k} \sim P_{d^\star}$ is $s_y^2\, \mathbb{E}_{d^\star}\|g^\star(\mathbf{k}') - g^\star_{d^\star}(\mathbf{k}')\|^2 = s_y^2\, \delta_{d^\star}(Q)^2$ by Definition 2, which is Eq. (4).

The left side of Eq. (4) is the excess risk of $\hat{\mathbf{y}}$ under squared loss. For any predictor $f(\mathbf{k})$ with finite second moment the tower property removes the cross term in $\mathbb{E}\|f(\mathbf{k}) - \mathbf{y}\|^2 = \mathbb{E}\|f(\mathbf{k}) - \mathbb{E}[\mathbf{y} \mid \mathbf{k}]\|^2 + \mathbb{E}\|\mathbf{y} - \mathbb{E}[\mathbf{y} \mid \mathbf{k}]\|^2$, and the second term, the Bayes risk of $d^\star$, does not depend on f. The same decomposition under Q shows that a predictor of $\mathbf{k}'$ alone is risk-optimal on the source pool if and only if it equals $g^\star$ Q-almost everywhere, so every such predictor has the target excess risk of Eq. (4). The argument does not use that $d^\star$ lies outside the source set; it applies to any domain d with Q the pool that contains it. □

The source domains. Applied to a source domain, the identity says how much a predictor could gain from knowing which domain a window comes from.

Corollary 1 (Source domains)**.** *For a source domain d with scales s_d and step scale $s_{y,d}$, and Q the pooled source distribution, $\mathbb{E}_{P_d}\|s_{y,d}\, g^\star(\phi_{s_d}(\mathbf{k})) - \mathbb{E}[\mathbf{y} \mid \mathbf{k}]\|^2 = s^2_{y,d}\, \delta_d(Q)^2$. A predictor that identifies the domain of a window, for instance from its absolute speed, and replaces $g^\star$ by $g^\star_d$ therefore lowers the risk on d by exactly $s^2_{y,d}\delta_d(Q)^2$, and no predictor lowers it by more.*

Proof. The identity is Proposition 1 applied to d, which its proof allows. Since $s_{y,d}\, g^\star_d(\phi_{s_d}(\mathbf{k})) = \mathbb{E}_{P_d}[\mathbf{y} \mid \mathbf{k}]$ is the Bayes predictor of d, its excess risk is zero, so the risk decomposition above gives the difference as $s^2_{y,d}\delta_d(Q)^2$, and no predictor has a smaller risk on d than the Bayes predictor. □

Implication (iv) rests on this. Within the training range the absolute speed of a window can identify its source domain, and the most that this identification can buy back is that domain's departure; outside the training range there is no source domain to identify.

A learned map. A predictor trained on finite source data applies a learned map $\hat{g}$ in place of $g^\star$, the estimation term of Section 3.2. Writing $\hat{g} - g^\star_{d^\star} = (\hat{g} - g^\star) + (g^\star - g^\star_{d^\star})$ and applying the triangle inequality in $L^2(P_{d^\star})$ in both directions places the root excess risk of $s_y\, \hat{g}(\phi_{s_{d^\star}}(\mathbf{k}))$ between $s_y\, |\delta_{d^\star}(Q) - e|$ and $s_y\, (\delta_{d^\star}(Q) + e)$, where $e = (\mathbb{E}_{d^\star}\|\hat{g}(\mathbf{k}') - g^\star(\mathbf{k}')\|^2)^{1/2}$ is the distance between $\hat{g}$ and $g^\star$ on the normalized inputs of $d^\star$; it involves $d^\star$ only through the distribution of those inputs, and equals the estimation error of $\hat{g}$ on the sources where the sources are dense. A learned predictor can therefore incur less than the ideal excess $s_y^2\delta^2_{d^\star}$, but only with $e > 0$, that is, by differing from $g^\star$ on inputs that the sources cover, so it is not risk-optimal on the sources and the sources alone cannot select it; against an in-domain reference that is itself learned rather than Bayes-optimal the measured excess is smaller still. This is why Section 5.3 checks the identity on the conditional means and asks of KINETRAJ only that its excess rank with $\hat{\delta}^2_d$.

An error in the supplied scales. The scales of the held-out domain are statistics of its observed windows (Section 5.1), so the predictor receives them with a relative error. For KINETRAJ it cannot reach the anchors, which use the raw window, and enters only through the gate and residual.

Proposition 2 (Scale error)**.** *Let the supplied scales be $\tilde{s}_d = (s_v(1+\epsilon_v),\, s_a(1+\epsilon_a),\, s_y(1+\epsilon_y))$ with $\epsilon = \max(|\epsilon_v|, |\epsilon_a|, |\epsilon_y|) < 1$, and let the backbone, the gate, and the residual head be Lipschitz with constants L_E, L_g (into the ℓ_1 norm of the weights), and L_r. Write $\mathbf{k}'_{\mathrm{m}}$ for the magnitude part of $\phi_{s_d}(\mathbf{k})$, $A_{\max}(\mathbf{k}) = \max_m \|A^{(m)}(\mathbf{k})\|$, and $\mathbf{r} = r_\theta(\mathbf{h}, \mathbf{e}_j)$. Then*

$$\big\|\hat{\mathbf{y}}(\mathbf{k}; \tilde{s}_d) - \hat{\mathbf{y}}(\mathbf{k}; s_d)\big\| \;\le\; \frac{2\epsilon}{1-\epsilon}\, L_E \|\mathbf{k}'_{\mathrm{m}}\| \big(L_g A_{\max}(\mathbf{k}) + (1+\epsilon)\, s_y L_r\big) + \epsilon\, s_y \|\mathbf{r}\|,$$

which is $O(\epsilon)$. The anchors do not contribute, and an error in s_y alone ($\epsilon_v = \epsilon_a = 0$) changes the hypothesis by $|\epsilon_y|\, s_y \|\mathbf{r}\|$ only.

Proof. Under $\tilde{s}_d$ each magnitude channel of the normalized input is multiplied by $(1+\epsilon_v)^{-1}$, $(1+\epsilon_a)^{-1}$, or $(1+\epsilon_v)/(1+\epsilon_a)$, all within $2\epsilon/(1-\epsilon)$ of one, so $\|\tilde{\mathbf{k}}' - \mathbf{k}'\| \le \frac{2\epsilon}{1-\epsilon}\|\mathbf{k}'_{\mathrm{m}}\|$ and $\|\tilde{\mathbf{h}} - \mathbf{h}\| \le L_E\|\tilde{\mathbf{k}}' - \mathbf{k}'\|$. The anchors are computed from the raw window and are unchanged, so the mixed anchor moves by at most $\|g(\tilde{\mathbf{h}}) - g(\mathbf{h})\|_1 A_{\max}(\mathbf{k}) \le L_g\|\tilde{\mathbf{h}} - \mathbf{h}\|A_{\max}(\mathbf{k})$. The residual term moves by $\|s_y(1+\epsilon_y)\, r_\theta(\tilde{\mathbf{h}}, \mathbf{e}_j) - s_y\, r_\theta(\mathbf{h}, \mathbf{e}_j)\| \le s_y(1+\epsilon)L_r\|\tilde{\mathbf{h}} - \mathbf{h}\| + \epsilon\, s_y\|\mathbf{r}\|$. Adding the two bounds gives the claim; when only s_y is in error, $\tilde{\mathbf{h}} = \mathbf{h}$ and only the last term remains. □

This is the scale term of Section 3.2. The sweep in Table 8 varies s_y alone at test time to assess the effect of residual scaling on KINETRAJ's ADE across six datasets.

Bounded losses. Proposition 1 concerns the conditional mean under squared loss, whereas the benchmark of Section 5 scores K hypotheses by the best of them, a bounded loss that depends on the whole conditional distribution rather than on its mean. The bound extends to such losses with the total-variation distance in place of the departure.

Remark 1 (Bounded losses). For any loss bounded by B, the best-of-K displacement error included, predicting $d^\star$ with the shared distribution Q in place of $Q_{d^\star}$ costs at most $2B$ times the mean conditional total-variation distance $\mathbb{E}_{\mathbf{k}'}[\mathrm{TV}(Q_{d^\star}(\cdot \mid \mathbf{k}'), Q(\cdot \mid \mathbf{k}'))]$, which is the distributional counterpart of $\delta_{d^\star}$.

Proof of Remark 1. For a loss $0 \le L \le B$ and distributions P, P', $|\mathbb{E}_P L - \mathbb{E}_{P'} L| \le B\,\mathrm{TV}(P, P')$. Fix $\mathbf{k}'$ and let h_Q and $h_{d^\star}$ be hypothesis sets that minimize the expected loss under $Q(\cdot \mid \mathbf{k}')$ and $Q_{d^\star}(\cdot \mid \mathbf{k}')$, assumed to exist. Then $\mathbb{E}_{Q_{d^\star}} L(h_Q) \le \mathbb{E}_Q L(h_Q) + B\,\mathrm{TV} \le \mathbb{E}_Q L(h_{d^\star}) + B\,\mathrm{TV} \le \mathbb{E}_{Q_{d^\star}} L(h_{d^\star}) + 2B\,\mathrm{TV}$, with $\mathrm{TV} = \mathrm{TV}(Q_{d^\star}(\cdot \mid \mathbf{k}'), Q(\cdot \mid \mathbf{k}'))$; averaging over $\mathbf{k}'$ under the target marginal gives the claim. □

This is also why implication (iii) is verified in Section 5.3 in the proposition's own terms, the squared error of a point prediction, rather than with the best-of-K error of Table 1. Under a bounded loss the departure enters only through a distributional distance, and the two need not order the folds alike.