DRIFTCF: GENERATIVE MODELING OF JOINT POTENTIAL OUTCOME DISTRIBUTIONS VIA ONE-STEP PARTICLE DRIFTING

Anonymous authors
Paper under double-blind review

ABSTRACT

Interventions often change several dependent outcomes, making the treatment-specific joint potential-outcome distribution the relevant object for uncertainty, dependence, and joint risk. Learning this law is difficult because each unit reveals only one factual outcome vector, while the missing counterfactual distribution must preserve its complete dependency structure. Existing expressive generators obtain samples through iterative denoising, numerical integration, or ordered generation. We introduce *Drifting-based CounterFactual generation* (DriftCF). Causal identification turns the target into conditional joint-distribution matching; DriftCF solves it by attracting complete generated vectors toward factual outcomes and repelling particles according to the model's own concentration. Detached regression amortizes this correction into a non-autoregressive generator that produces one complete vector in one forward pass. We establish the field's smoothed-score form, target fixed point, and amortized transport property. Experiments test joint-law recovery, dependence, the attraction–repulsion mechanism, and one-step efficiency on multi-outcome benchmarks and an online incentive-allocation study. Code is available in the Supplementary Material.

1 INTRODUCTION

An intervention rarely changes only one outcome. A treatment can jointly alter efficacy and adverse-event risk; an incentive can jointly affect engagement, revenue, and cost. Decisions therefore depend on joint events, not only separate conditional means or marginals. Distributional causal estimators broaden mean effects to uncertainty, while multi-outcome estimands retain selected aggregate information (Byambadalai et al., 2024; Melnychuk et al., 2024a; Kennedy et al., 2019); DIME directly targets the joint interventional outcome law (Ma et al., 2025).

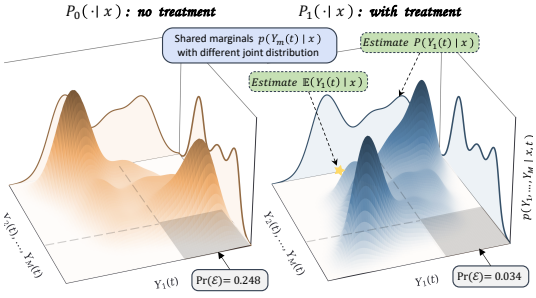


Figure 1: Identical outcome marginals can conceal a treatment-induced change in the joint law and hence in decision-relevant event probabilities.

Figure 1 isolates the resulting blind spot. In its sharpest form, treatment and control satisfy $\mathcal{L}(Y_m(1) \mid X = x) = \mathcal{L}(Y_m(0) \mid X = x)$ for every outcome m, while $\mathcal{L}(Y(1) \mid X = x) \neq \mathcal{L}(Y(0) \mid X = x)$. Every outcome-wise mean contrast is then zero, and even perfect marginal estimators cannot distinguish the arms, while correlation, tail dependence, and joint risk change. The scientific target must therefore move from points to distributions and from separate outcomes to their joint law. Let T denote treatment and $Y = (Y_1, \ldots, Y_M)$ the vector of outcomes. The object we seek for a person with covariates x is

$$P_t(\cdot \mid x) := \mathcal{L}\big(Y(t) \mid X = x\big), \qquad Y(t) = \big(Y_1(t), \ldots, Y_M(t)\big). \tag{1}$$

This law contains means and quantiles together with dependence and joint-event probabilities. Distinct copulas preserve every marginal while changing these decisions, so the model must generate coherent vectors rather than concatenate independent samples.

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.

Learning Equation equation 1 combines counterfactual missingness with dependence recovery. Each unit reveals only $Y_i(T_i)$, so another treatment's law must be identified from comparable factual units under consistency, exchangeability, and overlap (Rubin, 1974; Rosenbaum & Rubin, 1983). That transfer must recover all coordinates jointly; multiplying fitted marginals cannot reconstruct their coupling. The task is therefore causally identified *conditional joint-distribution matching*.

Existing causal generators inherit costly sampling paths. DiffPO uses diffusion for potential-outcome distributions, DIME combines diffusion with ordered multi-outcome generation, and interventional flows or PO-Flow evaluate an invertible or integrated path (Ma et al., 2024; 2025; Melnychuk et al., 2023; Wu et al., 2026). These models are expressive, but each query replays denoising, numerical integration, or sequential factorization (Figure 2).

Our central observation is that identification and particle drifting meet at conditional distribution matching. Gaussian drifting compares data and model through local mean-shift fields (Deng et al., 2026; Lai et al., 2026). Applied to complete outcome vectors, factual attraction moves particles toward the identified law and model repulsion removes excess concentration. Their difference therefore corrects density, diversity, and dependence without prescribing an inference-time path.

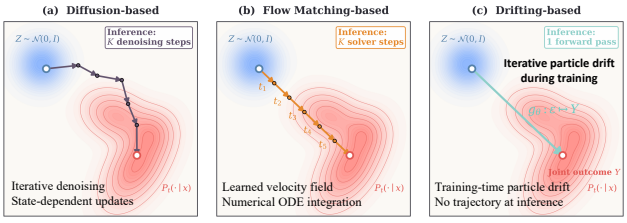


Figure 2: Diffusion and flow matching replay an iterative path for each draw, whereas drifting compiles training-time particle motion into one-pass inference.

DriftCF computes this field in treatment-aware factual neighborhoods and regresses detached particle displacements into a joint generator. Repeated training updates compile transport into the parameters; one noise draw then produces one complete potential-outcome vector. Algorithm 1 gives the data flow, while nuisance fitting and general treatment/outcome configurations are detailed in the appendix. Our contributions are:

- **Conditional drifting for joint potential outcomes.** We cast the identified interventional law as conditional joint-distribution matching and construct a treatment-aware attraction–repulsion field over complete outcome vectors.
- **One-step amortized generation.** We project particle corrections into a non-autoregressive generator whose shared latent draw produces all outcomes simultaneously, removing denoising, numerical integration, and outcome-wise recursion at inference.
- **Evidence from mechanism to decision.** We evaluate conditional drifting through oracle-field recovery and joint multi-outcome benchmarks, connecting complete-distribution recovery and one-step sampling efficiency to decision-relevant performance.

2 Preliminaries: Particle Drifting

Drifting evolves a generator's pushforward during training while preserving a direct noise-to-sample map at inference (Deng et al., 2026). Its essential object is a particle field that compares local mass under the data and model distributions. The central difficulty is to construct a corrective direction that moves an imperfect generator toward the data law without sending all particles to the same local mode. Given the Gaussian kernel $k_h(z, z') = \exp\left(-\frac{\|z-z'\|^2}{2h^2}\right)$, the kernel mean-shift field of a density R on $\mathbb{R}^d$ is defined as

$$M_R(z) = \frac{\int k_h(z, z')(z' - z)R(z')\,\mathrm{d}z'}{\int k_h(z, z')R(z')\,\mathrm{d}z'},$$

where the denominator is the kernel-smoothed density $\overline{R}_h(z) = \int k_h(z, z')R(z')\,\mathrm{d}z'$. For a Gaussian kernel, this local displacement has the exact score interpretation

$$M_R(z) = h^2 \nabla_z \log \overline{R}_h(z). \tag{2}$$

Thus Gaussian mean shift is the score of a kernel-smoothed density. This identity connects drifting to score models and to broader particle transport through kernel mean embeddings and MMD gradient

flows (Deng et al., 2026; Lai et al., 2026; Wang & Nüsken, 2024; Belhadji et al., 2025; Duong et al., 2025). For data distribution P and generator distribution Q_θ, define

$$V(z) = M_P(z) - M_{Q_\theta}(z) = h^2 \left[\nabla_z \log \overline{P}_h(z) - \nabla_z \log \overline{Q}_{\theta,h}(z)\right]. \tag{3}$$

The first term attracts a particle toward data mass; the second subtracts the mean shift induced by the model itself. Repulsion is therefore the correction that turns local attraction into a data–model score mismatch. It prevents attraction-only mode collapse and makes distributional agreement a fixed point. Appendix B develops the Gaussian-kernel connection to score-based mean shift, including its high-dimensional interpretation through concentration of pairwise kernel mass. Rather than numerically integrating V at inference, drifting constructs a detached training target

$$\widetilde{z} = \mathrm{sg}[z + \eta V(z)], \qquad \mathcal{L}(\theta) = \mathbb{E}\|z - \widetilde{z}\|_2^2. \tag{4}$$

This detached regression repeatedly changes the generator pushforward so that a direct output anticipates its transported location. The transport is iterative during training but amortized at inference. DriftCF conditions this primitive on a causally identified context and applies it to outcome vectors.

3 Problem Formulation

We observe $\mathcal{D} = \{(X_i, T_i, Y_i)\}_{i=1}^n$, where X_i is pre-treatment, T_i is the assigned treatment, and $Y_i = Y_i(T_i) \in \mathcal{Y}_1 \times \cdots \times \mathcal{Y}_M$ is the factual outcome vector. The target is the conditional law in Equation equation 1. The implementation supports

$$\mathcal{T} \in \big\{\{0,1\},\ \{1,\dots,K\},\ \{t_1 < \cdots < t_K\},\ \mathcal{I} \subseteq \mathbb{R}\big\},$$

corresponding to binary, unordered multi-valued, ordinal, and continuous treatments. We assume consistency/SUTVA, weak unconfoundedness $Y(t) \perp T \mid X$, positivity at the queried treatment (or dose neighborhood), pre-treatment covariates, and local smoothness for continuous treatment (Rubin, 1974; Rosenbaum & Rubin, 1983).

Proposition 3.1 (Identification of the joint interventional law)**.** *Under (A1)–(A3), for every supported (x, t),*

$$\mathcal{L}\big(Y(t) \mid X = x\big) = \mathcal{L}\big(Y \mid X = x, T = t\big). \tag{5}$$

For continuous treatment, the equality is between regular conditional laws, and (A5) permits local kernel estimation as the treatment bandwidth vanishes.

Because the equality holds on the product outcome space, it identifies the full within-treatment dependence, not only the marginals. Counterfactual generation is therefore reduced to matching an observed conditional joint law. Proofs of all formal results are collected in Appendix A in their order of appearance. Our model learns

$$q_\theta(y \mid x, t) \approx P_t(y \mid x), \qquad \widehat{Y}^{(s)}(t) \sim q_\theta(\cdot \mid x, t).$$

Every draw is a complete vector. Except under conditional independence, $\prod_{m=1}^M p\big(Y_m(t) \mid X = x\big) \neq p\big(Y_1(t), \dots, Y_M(t) \mid X = x\big)$, so independently accurate marginals can still give incorrect joint decisions. The estimand is the within-treatment law in Equation equation 1, rather than a cross-world coupling such as $\mathcal{L}(Y(t_1), Y(t_0) \mid X = x)$. This distinction keeps the scientific question aligned with what standard identification assumptions recover.

4 Drifting-based CounterFactual Generation

4.1 One-step joint generator

DriftCF represents each outcome sample as one particle. A fixed adapter maps heterogeneous outcomes into a continuous, decodable space,

$$Z = a(Y, U) \in \mathbb{R}^{d_Z}, \qquad U \perp (X, T, Y), \qquad d(a(Y, U)) = Y \text{ a.s.}$$

Given a frozen covariate representation $\phi(x)$ and a treatment representation $e_T(t)$, the generator produces

$$\widehat{Z}(t) = g_\theta\big(\epsilon, \phi(x), e_T(t)\big), \qquad \widehat{Y}(t) = d\big(\widehat{Z}(t)\big), \quad \epsilon \sim \mathcal{N}(0, I). \tag{6}$$

The joint pushforward of g_θ carries marginals and dependence simultaneously. Equation equation 6 has no outcome ordering and requires one generator evaluation per draw. The adapter construction, including its lossless decoding mechanism for mixed outcomes, is given in Appendix A.2.

4.2 CAUSAL CONTEXT AND TREATMENT MASKING

Within one generator fold, each observation follows two synchronized routes. Its pre-treatment covariates become $\phi(x_i)$ and its factual outcome vector becomes $z_i = a(y_i, u_i)$. Treatment also has two distinct roles: $e_T(t_i)$ conditions what the generator produces, whereas $k_T(t_i, t_j)$ determines which factual and generated particles may contribute to the query's drift. Binary and unordered treatments use exact same-arm masks; ordinal levels and continuous doses use local Gaussian weights on rank or standardized dose. The exact definitions and bandwidth limits are given in Appendix A.3.

For $c = (\phi(x), t)$, define

$$k_c(c, c') = k_x\big(\phi(x), \phi(x')\big) k_T(t, t'), \tag{7}$$

The product weight first applies the causal treatment mask and then measures covariate proximity. For query context c_i, the resulting neighbors and their factual vectors feed attraction; particles generated for the same weighted contexts feed repulsion. This is the fold-in data path from (X_i, T_i, Y_i) to V_{is}. The representation, adapter, and bandwidths are estimated on the complementary fold and frozen; they provide the fixed geometry used by the theoretical analysis rather than another stage of outcome generation.

4.3 CONDITIONAL ATTRACTION–REPULSION FIELD

Let $c = (\phi(x), t)$. The data and model joint laws are

$$P(c, z) = p(c)p(z \mid c), \qquad Q_\theta(c, z) = p(c)q_\theta(z \mid c). \tag{8}$$

They share $p(c)$ because the model conditions on context rather than generating it. Matching the joint law is therefore equivalent to matching $p(z \mid c)$ on supported contexts.

The practical challenge is that each context supplies only one factual vector, whereas distribution matching needs a direction for every generated particle. DriftCF estimates that direction by comparing two local centers in the *complete-vector* space: the center supported by factual outcomes and the center already induced by model particles. The first pulls unsupported samples toward data; subtracting the second prevents this pull from concentrating every sample at the same mode, as illustrated in Figure 3. For a batch of size B, encode $z_j = a(y_j, u_j)$ and draw S particles per context, $\widehat{z}_{is} = g_\theta(\epsilon_{is}, \phi(x_i), e_T(t_i))$. Factual vectors produce attraction

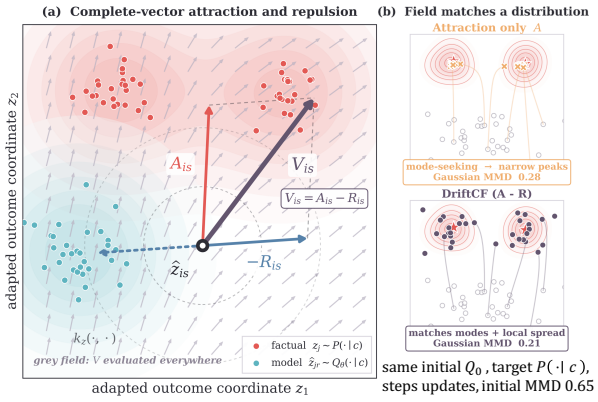


Figure 3: Conditional attraction–repulsion field, preserving local spread while matching the target distribution.

$$A_{is} = \frac{\sum_{j=1}^{B} k_c(c_i, c_j) k_z(\widehat{z}_{is}, z_j)(z_j - \widehat{z}_{is})}{\sum_{j=1}^{B} k_c(c_i, c_j) k_z(\widehat{z}_{is}, z_j) + \varepsilon_0}, \tag{9}$$

and model particles produce leave-one-out mean shift

$$R_{is} = \frac{\sum_{j=1}^{B} \sum_{r=1}^{S} \mathbf{1}[(j, r) \neq (i, s)] k_c(c_i, c_j) k_z(\widehat{z}_{is}, \widehat{z}_{jr})(\widehat{z}_{jr} - \widehat{z}_{is})}{\sum_{j=1}^{B} \sum_{r=1}^{S} \mathbf{1}[(j, r) \neq (i, s)] k_c(c_i, c_j) k_z(\widehat{z}_{is}, \widehat{z}_{jr}) + \varepsilon_0}. \tag{10}$$

The field is $V_{is} = A_{is} - R_{is}$. Because k_z measures distances between vectors rather than coordinate-wise distances, a wrong covariance or mode coupling changes local neighborhoods and hence changes the field even when all one-dimensional marginals are correct. This is the mechanism that turns complete-vector generation from an architectural preference into a distributional correction principle. We quantify local support using the inverse Herfindahl index

$$n_{\text{eff}}(x, t) = \frac{\Big(\sum_j k_c((\phi(x), t), c_j)\Big)^2}{\sum_j k_c((\phi(x), t), c_j)^2}. \tag{11}$$

Small n_{eff} signals a high-variance local field and is reported rather than hidden by extrapolation.

4.4 STOP-GRADIENT AMORTIZATION

We form the detached target

$$\widetilde{z}_{is} = \mathrm{sg}[\widehat{z}_{is} + \eta\,\mathrm{clip}(V_{is}, v_{\max})]\,, \qquad \mathcal{L}_{\mathrm{DriftCF}}(\theta) = \frac{1}{BS}\sum_{i=1}^{B}\sum_{s=1}^{S}\|\widehat{z}_{is} - \widetilde{z}_{is}\|_2^2. \tag{12}$$

Equation equation 12 detaches encoded outcomes, kernel weights, the field, and targets; the frozen context representation receives no gradient. This is a semi-gradient update: the field is recomputed after each parameter step, but its derivative is not taken within that step.

Proposition 4.1 (Amortized drift is a projected transport step)**.** *Let $u_{is} = \mathrm{clip}(V_{is}, v_{\max})$ and $J_{is} = \nabla_\theta g_\theta(\epsilon_{is}, c_i)$. Conditional on a sampled batch and its detached field,*

$$\nabla_\theta \mathcal{L}_{\mathrm{DriftCF}} = -\frac{2\eta}{BS}\sum_{i,s} J_{is}^\top u_{is}. \tag{13}$$

An infinitesimal gradient step therefore changes the stacked generated particles by $\delta\widehat{\mathbf{z}} = \frac{2\alpha\eta}{BS}\,\mathbf{K}_\theta\mathbf{u} + o(\alpha), \mathbf{K}_\theta = \mathbf{J}\mathbf{J}^\top \succeq 0$. Consequently, $\lim_{\alpha\downarrow 0}\frac{1}{\alpha}\langle \mathbf{u}, \delta\widehat{\mathbf{z}}\rangle = \frac{2\eta}{BS}\|\mathbf{J}^\top\mathbf{u}\|_2^2 \geq 0$. If the clipped population field lies in the closure of the generator tangent space, a zero population semi-gradient implies $V = 0$ and hence reaches the distributional fixed point characterized below.

Optimization therefore projects particle-space corrections onto motions realizable by the generator and compiles them into g_θ. Iteration occurs in parameter learning, whereas a trained sample requires only Equation equation 6. The proof is in Appendix A.

4.5 THEORETICAL GUARANTEES

Let $\overline{P}_h$ and $\overline{Q}_{\theta,h}$ denote smoothing of Equation equation 8 by the fixed kernel $k_c(c,c')k_z(z,z')$. The following result states why attraction–repulsion targets the full conditional law rather than only local diversity. The theorem conditions on a fixed context geometry. In implementation, every branch obtains its adapter, covariate representation, causal mask, and bandwidths from the complementary folds and freezes them before generator training. Cross-fitting therefore supplies the fixed smoothing operator required below; Algorithm 2 gives the complete nuisance and branch construction.

Appendix Figure 6 visualizes the Gaussian identity behind the theorem by comparing the mean-shift field with the scaled score of the smoothed density.

Lemma 4.2 (Conditional Gaussian mean shift)**.** *For a Gaussian outcome kernel with bandwidth h_z, the normalized population mean shift of any context–outcome law R equals $h_z^2\nabla_z \log \overline{R}_h(c,z)$ wherever the smoothed density is positive.*

Lemma 4.3 (Zero score mismatch)**.** *Under the positivity, connected-support, shared-context-marginal, and injective-smoothing assumptions of Theorem 4.4, equality of the smoothed data and model outcome scores implies $p(z \mid c) = q_\theta(z \mid c)$ almost everywhere.*

Theorem 4.4 (Conditional score correspondence and fixed point)**.** *For Gaussian k_z with bandwidth h_z, the population field satisfies*

$$V(c,z) = h_z^2\left[\nabla_z \log \overline{P}_h(c,z) - \nabla_z \log \overline{Q}_{\theta,h}(c,z)\right]. \tag{14}$$

Assume that P and Q_θ share the context marginal, their smoothed densities are positive on a connected outcome domain, the fixed smoothing operator is injective, and the model density is positive almost everywhere on that domain. If $V(c,z) = 0$ for $p(c)$-almost every c and Q_θ-almost every z, then

$$q_\theta(z \mid c) = p(z \mid c) \quad \textit{almost everywhere}.$$

Corollary 4.5 (Decoded causal target)**.** *Under Proposition 3.1, ODS in Proposition A.3, and the lossless adapter condition in Proposition A.1, the decoded zero-field distribution satisfies*

$$d_{\#}q_\theta^Z(\cdot \mid \phi(x), t) = \mathcal{L}\big(Y(t) \mid X = x\big) \quad \textit{for every supported } (x,t).$$

Corollary 4.6 (From a population stationary point to the causal target)**.** *For a frozen branch satisfying Proposition A.5, suppose the clipped population field lies in the closure of the generator tangent space. A zero population detached semi-gradient then implies a zero field, and the decoded distribution equals $\mathcal{L}(Y(t) \mid X = x)$ at supported contexts.*

Lemma 4.7 (Ideal score transport dissipates conditional KL). *For smooth positive conditional laws with vanishing boundary flux, the ideal velocity $v = \nabla_z \log p - \nabla_z \log q$ satisfies*

$$\frac{\mathrm{d}}{\mathrm{d}\tau} \mathrm{KL}(q_\tau(\cdot \mid c) \| p(\cdot \mid c)) = -\mathbb{E}_{q_\tau} \|v(Z, c)\|_2^2 \leq 0.$$

The same averaged descent holds after a self-adjoint positive-semidefinite tangent projection.

Theorem 4.4 establishes that repulsion subtracts the model score and places the complete target law at equilibrium. The first corollary connects this equilibrium to the decoded causal estimand; the second connects it to the amortized parameterization. Appendix A contains the statements' proofs, and Appendix B develops the score interpretation.

4.6 Algorithm overview

Algorithm 1 DriftCF training and one-step inference

Require: Training data $\mathcal{D}$, particle count S, step size η, clipping radius $v_{\max}$

1: Fit the outcome adapter and distribution-predictive covariate geometry on nuisance folds; freeze both.
2: Map factual Y_i to joint particles z_i and X_i to $\phi(x_i)$; map T_i separately to generator code $e_T(t_i)$ and causal mask k_T.
3: Draw noise and generate S complete particles $\widehat{z}_{is} = g_\theta(\epsilon_{is}, \phi(x_i), e_T(t_i))$.
4: Apply $k_c = k_x k_T$ to select supported factual and model neighbors.
5: Compute factual attraction A_{is} and leave-one-out model repulsion R_{is} on complete vectors.
6: Form $\widetilde{z}_{is} = \mathrm{sg}[\widehat{z}_{is} + \eta \,\mathrm{clip}(A_{is} - R_{is}, v_{\max})]$ and update the generator with Equation equation 12.

Inference: For query (x, t), draw noise, evaluate g_θ once, and decode one complete vector.

Algorithm 1 separates training-time particle correction from inference. One forward pass yields one complete joint outcome vector; repeating the same query with independent noise draws yields samples for means, quantiles, intervals, covariance, and joint-event probabilities. Cross-fitting, branch aggregation, mixed outcomes, treatment variants, and support checks are detailed in Algorithm 2.

5 Related Work

Causal estimands: from means to joint laws. Meta-learners, representation learners, doubly robust methods, and efficient adjustment theory primarily target means or selected causal functionals (Johansson et al., 2016; Shalit et al., 2017; Künzel et al., 2019; Curth & van der Schaar, 2021; Chernozhukov et al., 2018; Kennedy, 2023; Rotnitzky & Smucler, 2020; Feuerriegel et al., 2024). Distributional methods extend the target to potential-outcome laws, aleatoric treatment-effect uncertainty, and RKHS embeddings of conditional counterfactual distributions (Melnychuk et al., 2023; Ma et al., 2024; Byambadalai et al., 2024; Melnychuk et al., 2024a; Anancharoenkij & Ponnoprat, 2026). Multi-outcome aggregate estimands retain selected summaries (Kennedy et al., 2019), whereas DIME directly models a mixed-type joint interventional outcome law with masked conditional diffusion and autoregressive generation (Ma et al., 2025). DriftCF targets the same complete-vector object through simultaneous particle transport.

Generative paths for potential outcomes. Adversarial and latent-variable models provide flexible potential-outcome generators (Yoon et al., 2018; Louizos et al., 2017; Bica et al., 2020). Diffusion models learn cross-sectional or longitudinal counterfactual outcome distributions through iterative denoising (Ho et al., 2020; Song et al., 2021; Ma et al., 2024; 2025; Alinezhad et al., 2026); prior-fitted longitudinal models instead amortize across synthetic causal tasks but recursively roll out future outcomes (Zare et al., 2026). Continuous normalizing flows and flow matching use invertible transformations or integrated velocity fields (Melnychuk et al., 2023; Lipman et al., 2023; Wu et al., 2026), with recent analyses linking their rates to kernel density estimation and bounding distributional error through flow-matching loss (Kunkel & Trabs, 2025; Su et al., 2025). PO-Flow is the closest flow-based generator on the shared potential-outcome distribution task; its factual-conditioned encode–decode task addresses a cross-world object, as does retrospective prediction based on explicit cross-world dependence (Bodik, 2026). DriftCF instead amortizes conditional endpoint correction into a simultaneous vector map for the identified within-treatment law.

Distribution matching, particle transport, and causal geometry. Kernel discrepancies compare distributions without an explicit likelihood (Gretton et al., 2012), while Stein descent, MMD gradient flows, kernel-mean-embedding transport, and MMD quantization convert distributional mismatch into interacting-particle motion (Liu & Wang, 2016; Hertrich et al., 2024; Wang & Nüsken, 2024; Belhadji et al., 2025; Duong et al., 2025). Drifting combines normalized data attraction and model repulsion and amortizes the field into one-step generation (Deng et al., 2026); its Gaussian mean shifts admit a smoothed-score interpretation (Lai et al., 2026). DriftCF brings this mechanism to an identified conditional joint law. The causal extension fixes which factual observations support a query: treatment encoding conditions the generator, treatment masking controls borrowing, and an outcome-predictive representation defines the covariate metric. This role differs from mean-oriented balancing or orthogonal representations (Johansson et al., 2016; Shalit et al., 2017; Hansen, 2008; Melnychuk et al., 2026); representation compression and jointly learned kernels can otherwise alter the estimand or neighborhood geometry (Christgau et al., 2024; Melnychuk et al., 2024b; Clivio et al., 2026; Schulte et al., 2025; Ober et al., 2021; Liu et al., 2020; Kübler et al., 2020). Appendix C gives the full treatment and nuisance construction.

6 Experiments

6.1 Setup

Datasets. We use four complementary benchmarks. ACIC and MIMIC-III evaluate binary-treatment joint-distribution recovery under the semi-synthetic DIME protocol; IHDP evaluates heterogeneous treatment effects with mixed outcomes; and News evaluates continuous-treatment response. All four use real covariates with known multi-outcome potential-outcome mechanisms. For MIMIC-III, we follow DIME's semi-synthetic construction based on high-dimensional clinical covariates, mechanical ventilation treatment, and mixed length-of-stay and mortality-related outcomes. Complete construction, splits, and outcome definitions are in Appendix E.

Baselines. We compare with DIME, DiffPO marginal product, PO-Flow, joint conditional diffusion, and joint conditional flow matching. A * denotes an adaptation to a treatment regime or joint-distribution output not supported by the original method. All methods use the same splits, tuning budget, evaluation draws, and outcome normalization; Appendix E specifies the adaptations and treatment-specific benchmark results.

Metrics. We report the empirical Wasserstein-1 distance (W_1) for the DIME-aligned binary benchmarks and multivariate energy distance for IHDP and News (Gneiting & Raftery, 2007). The corresponding samplewise energy score is

$$\widehat{S}_{\mathrm{ES}}(q, y) = \frac{1}{S}\sum_{s} \|\xi_s - y\|_2 - \frac{1}{2S^2}\sum_{s,s'} \|\xi_s - \xi_{s'}\|_2, \tag{15}$$

where $\xi_s \sim q$ are model draws and y is a ground-truth potential-outcome vector. Averaging this strictly proper score over target draws yields the empirical multivariate energy distance up to a target-only constant. Because marginal metrics cannot measure whether outcomes are correctly coupled, dependence error averages absolute covariance and Spearman rank-correlation errors across all outcome pairs. Appendix E gives the treatment-specific benchmark breakdowns.

Implementation. Reproduction details, baseline adaptations, per-treatment results, and ablations are deferred to Appendix E and supplementary material.

6.2 Joint distribution recovery

Table 1 is the central test of distribution recovery. DIME anchors the native joint-distribution comparison, DiffPO marginal product tests whether accurate marginals recover a joint law, and joint diffusion and flow matching compare DriftCF with alternative joint generators. The dependence column isolates the information that marginal accuracy cannot identify.

We isolate the proposed update on a controlled low-dimensional task with known conditional density and population endpoint field. The task varies mode imbalance, heteroskedasticity, local support, and treatment-dependent covariance. All endpoint methods use identical particles, kernels, batches, generators, and training compute.

6.3 Attraction–repulsion mechanism

Table 1: Joint potential-outcome distribution recovery. ACIC and semi-synthetic MIMIC report Wasserstein-1 distance ($W_1 \downarrow$); IHDP and News report multivariate energy distance ($\downarrow$). Dependence ($\downarrow$) averages covariance and rank-correlation error across outcome pairs and benchmarks. * denotes an adaptation.

Method	ACIC	MIMIC	IHDP	News	Dependence
DIME*	1.02(0.09)	1.64(0.11)	0.82(0.06)	3.12(0.23)	0.12
DiffPO marginal product*	1.26(0.14)	1.92(0.17)	0.89(0.12)	2.71(0.36)	0.16
PO-Flow*	0.95(0.15)	1.77(0.23)	0.44(0.04)	2.65(0.25)	0.11
Joint conditional diffusion	2.25(0.32)	2.34(0.31)	0.77(0.09)	2.54(0.14)	0.18
Joint conditional flow matching	1.45(0.17)	2.08(0.22)	0.59(0.13)	2.77(0.08)	0.19
DriftCF	0.87(0.12)	1.29(0.14)	0.29(0.05)	2.36(0.18)	0.07

Table 2: Controlled mechanism evaluation against the known population endpoint field. Flow matching learns a path velocity, so endpoint-field metrics are not applicable.

Training direction	Field MSE $\downarrow$	Cosine $\uparrow$	Mean error $\downarrow$	Mode mass $\downarrow$	Dependence $\downarrow$
Attraction only	1.82(0.19)	0.68(0.02)	0.23(0.02)	0.32(0.04)	0.17
Cond-MMD grad	1.76(0.12)	0.71(0.01)	0.14(0.03)	0.13(0.02)	0.21
Cond-flow	n/a	n/a	0.11(0.02)	0.11(0.01)	0.19
DriftCF	1.54(0.13)	0.78(0.02)	0.08(0.02)	0.08(0.02)	0.07

Table 2 separates mean localization from recovery of the complete law (Avg. 4 datasets). Attraction-only can move particles toward data modes, but only model mean-shift subtraction corrects excess concentration in the endpoint field. Field agreement, mode mass, and dependence therefore test whether repulsion is the distributional correction predicted by Equation equation 3, rather than an auxiliary diversity regularizer. Repulsion-only, self-included repulsion, objective controls, and field visualizations are reported in Appendix E.

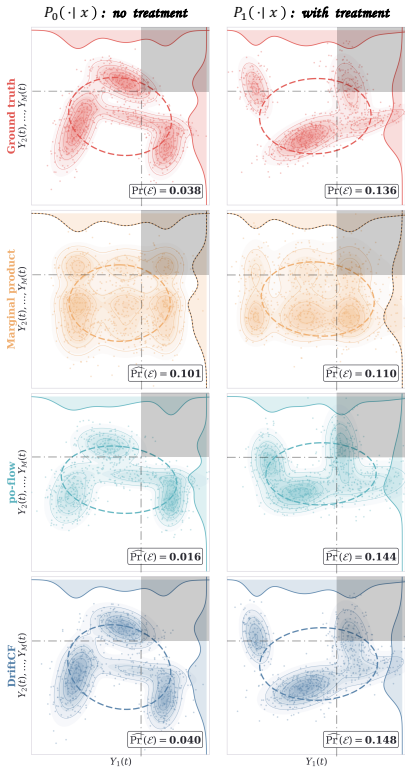


Figure 4: Held-out joint outcome distributions at representative incentive intensities on real industrial data.

6.4 Ablations

Table 3 reports the matched-objective and component ablations. Every row changes only the named component and reports the metric family affected by that component. The attraction-only row is the principal negative control for model-score subtraction, while repulsion-only and self-included repulsion separate data anchoring from diversity. Independent outcome heads isolate complete-vector generation. Continuous-dose treatment controls test local smoothing against no borrowing and exact matching. Encoder rows test outcome-distribution sufficiency and cross-fitting, and the adapter row tests whether coordinate-wise handling loses decoded joint structure.

6.5 Industrial Deployment

We deploy DriftCF on a large-scale internet platform with continuous incentive intensity as treatment and the mixed outcome vector $Y = (\text{cost}, \text{ watching time}, \text{ DAU}, \text{ revenue})$. The four outcomes combine nonnegative continuous, binary/count, and heavy-tailed monetary variables and exhibit strong treatment-dependent dependence. High-dimensional numerical covariates make this the primary evaluation of the discriminative encoder and local continuous-treatment geometry.

We evaluate online value in a user-randomized three-arm experiment comprising the point-estimate policy, a point-estimate policy, and the DriftCF distributional policy under identical eligibility, candidate incentives, budget accounting, and serving constraints. The deployed DriftCF policy uses joint potential-outcome samples over the four offline outcomes to optimize global incentive alloca-

Table 3: Ablations.

Axis	Variant	Energy ↓	Dependence ↓	Field cosine ↑	Norm n_{eff} ↑
Objective	Conditional MMD gradient	1.25	0.13	0.74	0.83
Objective	Conditional energy	1.48	0.13	0.66	0.74
Objective	Conditional flow matching	1.17	0.13	0.73	0.87
Field	Attraction only	1.34	0.18	0.71	0.78
Field	Repulsion only	1.39	0.20	0.72	0.75
Field	Include self in repulsion	1.19	0.09	0.75	0.94
Generator	Independent outcome heads	1.75	0.19	0.68	0.72
Generator	Autoregressive output	1.19	0.11	0.72	0.89
Treatment	No treatment kernel	2.14	0.23	0.54	0.57
Treatment	Exact matching for continuous dose	1.45	0.19	0.72	0.84
Encoder	Mean-score probe	1.22	0.12	0.76	0.82
Encoder	Jointly updated geometry	1.16	0.14	0.68	0.77
Encoder	Same-fold nuisance fit	1.18	0.17	0.67	0.67
Adapter	Independent outcome adapters	2.23	0.26	n/a	0.54
DriftCF	**Full model**	**1.14**	**0.07**	**0.78**	**1**

Table 4: Online business impact after deployment. Entries are relative lifts over the point-estimate policy with randomization-unit-level p-values; bold indicates statistical significance under the predeclared test.

Comparison	DAU	Watching Time	Video View Cnt	Ad Revenue
DriftCF vs. point-estimate policy	0.021% ($p = 0.31$)	0.334% ($p < 0.001$)	0.267% ($p = 0.01$)	0.504% ($p = 0.03$)

tion. Table 4 reports the predeclared comparison with the point-estimate policy as relative lifts and randomization-unit-level p-values.

Figure 4 shows that DriftCF tracks treatment-dependent coupling among cost, watching time, DAU, and revenue, whereas the marginal-product model assigns mass to unsupported outcome combinations. These complete-vector samples are used by the deployed policy to evaluate candidate incentive intensities under utility and risk constraints. Figure 5 compares joint-distribution quality with measured inference cost under the serving configuration. Iterative baselines vary diffusion steps, autoregressive calls, or ODE tolerance; DriftCF generates each complete vector with one network evaluation. The resulting quality–cost frontier provides the efficiency comparison without a separate latency table. Together, Table 1, Figures 4–5, and Table 4 connect offline joint-law recovery to serving efficiency and statistically significant online gains.

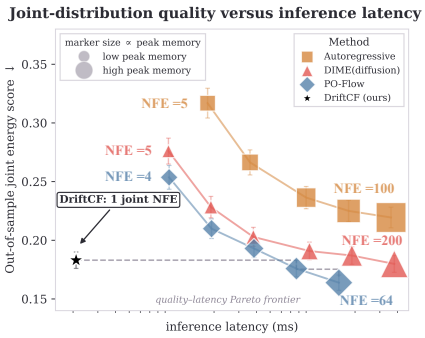


Figure 5: Joint-distribution quality versus inference cost under the industrial serving configuration.

7 Conclusion

DriftCF targets the treatment-specific joint potential-outcome law by combining a treatment-aware attraction–repulsion field over complete outcome vectors with detached regression that amortizes transport into one non-autoregressive generator pass. The field is a smoothed data–model score mismatch whose fixed point recovers the identified conditional law under the stated assumptions, while tangent coverage links a zero population semi-gradient to that fixed point. Across public semi-synthetic benchmarks and real industrial continuous-treatment data, the experiments evaluate joint-law and dependence recovery, matched-objective and component effects, one-step sampling efficiency, and decision value in incentive allocation. Together, the results show that conditional particle drifting provides a principled and efficient route from factual multi-outcome observations to coherent treatment-specific counterfactual distributions.

AI USE STATEMENT

Generative AI was used for LaTeX formatting, language polishing, and to assist in organizing the released source code for future open-source verification, usability, and reproduction of the reported model and results. The authors remain responsible for verifying every technical statement, proof, experimental result, final formulation, and for the correctness of the original code implementation.

REFERENCES

Farbod Alinezhad, Jianfei Cao, Gary J. Young, and Brady Post. Causal diffusion models for counterfactual outcome distributions in longitudinal data, 2026.

Thatchanon Anancharoenkij and Donlapark Ponnoprat. Conditional counterfactual mean embeddings: Doubly robust estimation and learning rates, 2026.

Ayoub Belhadji, Daniel Sharp, and Youssef Marzouk. Weighted quantization using MMD: From mean field to mean shift via gradient flows, 2025.

Ioana Bica, James Jordon, and Mihaela van der Schaar. Estimating the effects of continuous-valued interventions using generative adversarial networks. In *Advances in Neural Information Processing Systems*, volume 33, 2020.

Juraj Bodik. Retrospective counterfactual prediction by conditioning on the factual outcome: A cross-world approach, 2026.

Undral Byambadalai, Tatsushi Oka, and Shota Yasui. Estimating distributional treatment effects in randomized experiments: Machine learning for variance reduction. In *Proceedings of the 41st International Conference on Machine Learning*, 2024.

Victor Chernozhukov, Denis Chetverikov, Mert Demirer, Esther Duflo, Christian Hansen, Whitney Newey, and James Robins. Double/debiased machine learning for treatment and structural parameters. *The Econometrics Journal*, 21(1):C1–C68, 2018. doi: 10.1111/ectj.12097.

Alexander M. Christgau, Anton Rask Lundborg, and Niels Richard Hansen. Efficient adjustment for complex covariates: Gaining efficiency with DOPE, 2024.

Oscar Clivio, Alexander D'Amour, Alexander Franks, David Bruns-Smith, Chris Holmes, and Avi Feller. Deconfounding scores and representation learning for causal effect estimation with weak overlap. In *Proceedings of the International Conference on Artificial Intelligence and Statistics*, 2026.

Alicia Curth and Mihaela van der Schaar. On inductive biases for heterogeneous treatment effect estimation. In *Advances in Neural Information Processing Systems*, volume 34, 2021.

Mingyang Deng, He Li, Tianhong Li, Yilun Du, and Kaiming He. Generative modeling via drifting, 2026.

Richard Duong, Nicolaj Rux, Viktor Stein, and Gabriele Steidl. Wasserstein gradient flows of maximum mean discrepancy functionals with distance kernels under sobolev regularization. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 383(2298), 2025. doi: 10.1098/rsta.2024.0243.

Stefan Feuerriegel, Dennis Frauen, Valentyn Melnychuk, Jonas Schweisthal, Konstantin Hess, Alicia Curth, Stefan Bauer, Niki Kilbertus, Isaac S. Kohane, and Mihaela van der Schaar. Causal machine learning for predicting treatment outcomes. *Nature Medicine*, 30:958–968, 2024. doi: 10.1038/s41591-024-02902-1.

Tilmann Gneiting and Adrian E. Raftery. Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association*, 102(477):359–378, 2007. doi: 10.1198/016214506000001437.

Arthur Gretton, Karsten M. Borgwardt, Malte J. Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. *Journal of Machine Learning Research*, 13:723–773, 2012.

Ben B. Hansen. The prognostic analogue of the propensity score. *Biometrika*, 95(2):481–488, 2008.

Johannes Hertrich, Christian Wald, Fabian Altekrüger, and Paul Hagemann. Posterior sampling based on gradient flows of the MMD with negative distance kernel. In *International Conference on Learning Representations*, 2024.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, volume 33, pp. 6840–6851, 2020.

Fredrik D. Johansson, Uri Shalit, and David Sontag. Learning representations for counterfactual inference. In *Proceedings of the 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pp. 3020–3029, 2016.

Edward H. Kennedy. Towards optimal doubly robust estimation of heterogeneous causal effects. *Electronic Journal of Statistics*, 17(2), 2023.

Edward H. Kennedy, Shreya Kangovi, and Nandita Mitra. Estimating scaled treatment effects with multiple outcomes. *Statistical Methods in Medical Research*, 28(4), 2019.

Jonas M. Kübler, Wittawat Jitkrittum, Bernhard Schölkopf, and Krikamol Muandet. Learning kernel tests without data splitting. In *Advances in Neural Information Processing Systems*, 2020.

Lea Kunkel and Mathias Trabs. On the minimax optimality of flow matching through the connection to kernel density estimation, 2025.

Sören R. Künzel, Jasjeet S. Sekhon, Peter J. Bickel, and Bin Yu. Metalearners for estimating heterogeneous treatment effects using machine learning. *Proceedings of the National Academy of Sciences*, 116(10):4156–4165, 2019. doi: 10.1073/pnas.1804597116.

Chieh-Hsin Lai, Bac Nguyen, Naoki Murata, Yuhta Takida, Toshimitsu Uesaka, Yuki Mitsufuji, Stefano Ermon, and Molei Tao. A unified view of score-based and drifting models, 2026.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. In *International Conference on Learning Representations*, 2023.

Feng Liu, Wenkai Xu, Jie Lu, Guangquan Zhang, Arthur Gretton, and Danica J. Sutherland. Learning deep kernels for non-parametric two-sample tests. In *Proceedings of the 37th International Conference on Machine Learning*, 2020.

Qiang Liu and Dilin Wang. Stein variational gradient descent: A general purpose bayesian inference algorithm. In *Advances in Neural Information Processing Systems*, volume 29, 2016.

Christos Louizos, Uri Shalit, Joris M. Mooij, David Sontag, Richard Zemel, and Max Welling. Causal effect inference with deep latent-variable models. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

Yuchen Ma, Valentyn Melnychuk, Jonas Schweisthal, and Stefan Feuerriegel. DiffPO: A causal diffusion model for learning distributions of potential outcomes. In *Advances in Neural Information Processing Systems*, volume 37, 2024.

Yuchen Ma, Jonas Schweisthal, Hengrui Zhang, and Stefan Feuerriegel. A diffusion-based method for learning the multi-outcome distribution of medical treatments. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2025. doi: 10.1145/3711896.3736819.

Valentyn Melnychuk, Dennis Frauen, and Stefan Feuerriegel. Normalizing flows for interventional density estimation. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, 2023.

Valentyn Melnychuk, Stefan Feuerriegel, and Mihaela van der Schaar. Quantifying aleatoric uncertainty of the treatment effect: A novel orthogonal learner. In *Advances in Neural Information Processing Systems*, volume 37, 2024a.

Valentyn Melnychuk, Dennis Frauen, and Stefan Feuerriegel. Bounds on representation-induced confounding bias for treatment effect estimation. In *International Conference on Learning Representations*, 2024b.

Valentyn Melnychuk, Dennis Frauen, Jonas Schweisthal, and Stefan Feuerriegel. Orthogonal representation learning for estimating causal quantities. In *Proceedings of the International Conference on Artificial Intelligence and Statistics*, 2026.

Sebastian W. Ober, Carl Edward Rasmussen, and Mark van der Wilk. The promises and pitfalls of deep kernel learning. In *Proceedings of the Thirty-Seventh Conference on Uncertainty in Artificial Intelligence*, 2021.

Paul R. Rosenbaum and Donald B. Rubin. The central role of the propensity score in observational studies for causal effects. *Biometrika*, 70(1), 1983.

Andrea Rotnitzky and Ezequiel Smucler. Efficient adjustment sets for population average causal treatment effect estimation in graphical models. *Journal of Machine Learning Research*, 2020.

Donald B. Rubin. Estimating causal effects of treatments in randomized and nonrandomized studies. *Journal of Educational Psychology*, 66(5):688–701, 1974. doi: 10.1037/h0037350.

Robert Schulte, David Rügamer, and Thomas Nagler. Adjustment for confounding using pre-trained representations. In *Proceedings of the 42nd International Conference on Machine Learning*, 2025.

Uri Shalit, Fredrik D. Johansson, and David Sontag. Estimating individual treatment effect: Generalization bounds and algorithms. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 3076–3085, 2017.

Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2021.

Maojiang Su, Jerry Yao-Chieh Hu, Sophia Pi, and Han Liu. On flow matching KL divergence, 2025.

Linfeng Wang and Nikolas Nüsken. Measure transport with kernel mean embeddings, 2024.

Dongze Wu, David I. Inouye, and Yao Xie. Flow-based generative modeling of potential outcomes and counterfactuals. In *2026 IEEE International Symposium on Information Theory*, 2026. doi: 10.48550/arXiv.2505.16051. Accepted; arXiv:2505.16051.

Jinsung Yoon, James Jordon, and Mihaela van der Schaar. GANITE: Estimation of individualized treatment effects using generative adversarial nets. In *International Conference on Learning Representations*, 2018.

Amirhossein Zare, Amirhessam Zare, Herlock Rahimi, Reza Salarikia, and Mohammad Kashkooli. Causal longitudinal prior-fitted networks for counterfactual outcome prediction, 2026.

A Proofs

A.1 Proof of Proposition 3.1: Identification of the Joint Interventional Law

Proof roadmap. The argument has three steps. First, weak unconfoundedness makes treatment assignment irrelevant to the conditional law of $Y(t)$ once X is fixed. Second, consistency replaces $Y(t)$ by the observed vector Y among units receiving t. Third, positivity guarantees that this factual conditional law is defined on the queried support. Taking arbitrary measurable sets in the *joint* outcome space preserves all within-treatment dependence.

Assumptions and notation. Let $Y(t) = (Y_1(t), \ldots, Y_M(t))$ take values in the measurable product space $(\mathcal{Y}, \mathcal{B}(\mathcal{Y}))$. We use (A1) consistency, (A2) $Y(t) \perp T \mid X$, and (A3) positivity from Section 3. All equalities below are equalities between versions of regular conditional probabilities at supported (x, t).

Proof of Proposition 3.1. Fix an arbitrary $B \in \mathcal{B}(\mathcal{Y})$. By weak unconfoundedness,

$$\Pr(Y(t) \in B \mid X = x) = \Pr(Y(t) \in B \mid X = x, T = t).$$

Consistency gives $Y = Y(t)$ almost surely on $\{T = t\}$ and therefore

$$\Pr(Y(t) \in B \mid X = x, T = t) = \Pr(Y \in B \mid X = x, T = t).$$

Combining the two displays proves Equation equation 5. Positivity ensures that the factual conditional law is available at the queried (x, t). Because B was arbitrary in the product outcome space rather than restricted to coordinate-wise sets, the equality identifies the complete vector law, including covariance, copula structure, and joint-event probabilities.

For continuous treatment, conditioning is not an event-wise operation because $\Pr(T = t) = 0$. Let $f_{T|X}(t \mid x)$ denote the generalized propensity density. Positivity requires this density to be bounded away from zero in a neighborhood of the queried dose, extending the support role of propensity scores from discrete treatments (Rosenbaum & Rubin, 1983); Equation equation 5 is interpreted through regular conditional distributions. Assumption (A5) supplies local smoothness; under the usual kernel regime in which the relevant bandwidths vanish while the local effective sample size diverges, a local conditional-law estimator can remove smoothing bias without losing consistency to local variance. This estimation qualification does not alter the identification equality itself. □

A.2 Proof of Proposition A.1: Mixed-Outcome Preservation

Proposition A.1 (Adapter preservation)**.** *For any conditioning context c, if $d(a(Y, U)) = Y$ almost surely and $q_\theta^Z(\cdot \mid c) = p^Z(\cdot \mid c)$, then $d_\# q_\theta^Z(\cdot \mid c) = p^Y(\cdot \mid c)$.*

Proof roadmap. We first specify coordinate-wise adapters with deterministic decoders. We then view the adapted target law as the pushforward of (Y, U) through a. Finally, equality of adapted laws is transported through the decoder d; the almost-sure identity $d(a(Y, U)) = Y$ returns the original mixed-outcome law.

Assumptions and notation. The auxiliary noise U is independent of (X, T, Y). We use an augmented bijective construction: a maps (Y, U) into a continuous representation from which the outcome coordinate is recovered exactly by a measurable decoder d, so $d(a(Y, U)) = Y$ almost surely. The auxiliary coordinate supplies dequantization for discrete outcomes without changing the decoded outcome law. For a measurable map f and measure μ, $f_\#\mu(B) = \mu(f^{-1}(B))$ denotes the pushforward measure.

Proof of Proposition A.1. Fix a supported context c and any $B \in \mathcal{B}(\mathcal{Y})$. Equality of the adapted conditional laws and the definition of pushforward give

$$\begin{aligned}
(d_\# q_\theta^Z)(B \mid c) &= q_\theta^Z(d^{-1}(B) \mid c) \\
&= p^Z(d^{-1}(B) \mid c) \\
&= \Pr(a(Y, U) \in d^{-1}(B) \mid c) \\
&= \Pr(d(a(Y, U)) \in B \mid c) \\
&= \Pr(Y \in B \mid c),
\end{aligned}$$

where the last equality follows from the almost-sure decoding identity. Since this holds for every measurable B, the two decoded conditional measures are equal. □

A.3 Proof of Proposition A.2: Treatment Geometry

The treatment kernel used by the fixed context geometry is

$$k_T(t,t') = \begin{cases} \mathbf{1}[t=t'], & \text{binary or unordered,} \\ \exp\left(-\dfrac{(r(t)-r(t'))^2}{2h_r^2}\right), & \text{ordinal,} \\ \exp\left(-\dfrac{(\widetilde{t}-\widetilde{t}')^2}{2h_t^2}\right), & \text{continuous,} \end{cases} \tag{16}$$

where $r(t)$ is ordinal rank and $\widetilde{t}$ is standardized dose.

Proposition A.2 (Treatment geometry and limits)**.** *For the kernels in Equation equation 16: (i) exact matching restricts both drift components to the queried arm; (ii) for ordinal treatment, $h_r \to 0^+$ recovers exact matching and $h_r \to \infty$ removes treatment from the neighborhood; and (iii) under twice differentiable continuous-treatment laws, regular local support, and bounded conditional second moments, symmetric dose smoothing has leading interior bias $O(h_t^2)$, while local effective sample size controls the variance of its self-normalized empirical field.*

Proof roadmap. We treat the three claims separately. Step 1 uses the zero pattern of the exact-match kernel. Step 2 takes pointwise bandwidth limits of the ordinal Gaussian kernel. Step 3 writes continuous-treatment smoothing as a ratio of two local kernel moments: symmetry eliminates first-order terms, while normalized-weight variance is governed by the inverse Herfindahl index.

Assumptions and notation. For the continuous-treatment claim, fix an interior query (x,t,z) and absorb the covariate and outcome kernels into vector-valued numerator and scalar denominator functions $G_1(s)$ and $G_0(s)$ at factual dose s. Assume that both are twice continuously differentiable near t, $G_0(t) > 0$, the relevant conditional second moments are finite, and the treatment kernel is symmetric with finite second moment. These are the standard local-constant kernel-regression conditions referred to in Proposition A.2.

Proof of Proposition A.2. **Step 1: exact matching.** For exact matching, $t_j \neq t_i$ implies $k_c(c_i,c_j) = 0$, so all cross-arm terms vanish from Equations equation 9 and equation 10.

Step 2: ordinal limits. For ordinal treatment, if $t \neq t'$, then $(r(t)-r(t'))^2 \geq 1$. Hence $h_r \to 0^+$ sends every cross-level weight to zero while retaining $k_T(t,t) = 1$. Conversely, $h_r \to \infty$ gives $k_T(t,t') \to 1$ for every fixed pair, so treatment rank no longer affects neighborhood construction.

Step 3: continuous-treatment bias. Let $K(u) = \exp(-u^2/2)$ and write a generic population field component as

$$F_h(t) = \frac{\int K((s-t)/h_t) G_1(s)\,\mathrm{d}s}{\int K((s-t)/h_t) G_0(s)\,\mathrm{d}s}.$$

After the change of variables $s = t + h_t u$, Taylor expansion gives, for $\ell \in \{0,1\}$,

$$\int K((s-t)/h_t) G_\ell(s)\,\mathrm{d}s = h_t \left[\mu_0 G_\ell(t) + \frac{h_t^2 \mu_2}{2} G_\ell''(t) + o(h_t^2)\right],$$

where $\mu_j = \int u^j K(u)\,\mathrm{d}u$ and the $O(h_t)$ term inside the brackets vanishes because $\mu_1 = 0$. Since $G_0(t) > 0$, expansion of the ratio yields

$$F_h(t) = \frac{G_1(t)}{G_0(t)} + O(h_t^2).$$

This applies component-wise to attraction and repulsion, proving the claimed interior smoothing bias.

For variance, condition on nonnegative empirical weights w_j and define $\bar{w}_j = w_j / \sum_l w_l$. If the weighted summands have conditional variance bounded by σ^2 and are conditionally independent, then

$$\mathrm{Var}\left(\sum_j \bar{w}_j \xi_j \,\middle|\, w_{1:n}\right) \leq \sigma^2 \sum_j \bar{w}_j^2 = \frac{\sigma^2}{n_{\mathrm{eff}}}.$$

Thus local effective sample size controls the variance scale of the self-normalized empirical field. At a treatment boundary, kernel symmetry is truncated and a first-order term can remain; the $O(h_t^2)$ statement therefore requires an interior query or boundary correction. □

A.4 Proof of Proposition A.3: Distribution Sufficiency

Proposition A.3 (Estimand preservation under ODS)**.** *If $Y \perp X \mid \phi(X), T$, then, for every supported (x, t),*

$$p\big(Y \mid \phi(X) = \phi(x), T = t\big) = p(Y \mid X = x, T = t) = P_t(\cdot \mid x).$$

If only mean sufficiency holds, the first distribution is generally a mixture of distinct X-conditional laws, even when its mean is correct.

Proof roadmap. First, ODS removes any additional dependence on X after conditioning on $(\phi(X), T)$. Second, Proposition 3.1 converts the resulting factual conditional law into the interventional target. Finally, conditioning over all covariates collapsed to the same representation value exposes the mixture induced when only the conditional mean is retained.

Assumptions and notation. All statements concern regular conditional laws on supported contexts. Let $B \in \mathcal{B}(\mathcal{Y})$ be arbitrary and write $\Phi = \phi(X)$. ODS is the conditional independence $Y \perp X \mid (\Phi, T)$.

Proof of Proposition A.3. By ODS, for almost every supported (x, t),

$$\Pr(Y \in B \mid X = x, T = t) = \Pr(Y \in B \mid \phi(X) = \phi(x), T = t).$$

Combining this identity with Proposition 3.1 proves

$$\Pr(Y \in B \mid \Phi = \phi(x), T = t) = \Pr(Y(t) \in B \mid X = x).$$

Because B is arbitrary, the corresponding conditional probability measures are equal.

If ϕ is only mean-sufficient, then the law available to a representation-space learner is

$$\Pr(Y \in B \mid \phi(X), T = t) = \mathbb{E}[\Pr(Y \in B \mid X, T = t) \mid \phi(X), T = t]\,. \tag{17}$$

It is a mixture over covariate values collapsed by ϕ. The set-wise discrepancy at (x, t) is

$$\Delta(B; x, t) = \Pr(Y \in B \mid X = x, T = t) - \mathbb{E}[\Pr(Y \in B \mid X, T = t) \mid \phi(X) = \phi(x), T = t].$$

Mean sufficiency can make the first moment of this discrepancy vanish while leaving variance, modes, and cross-outcome dependence changed. The discrepancy vanishes in special cases where the full conditional law is measurable with respect to the conditional mean, including a Bernoulli outcome or certain additive-noise location families. □

A.5 Proof of Proposition A.4: Representation Geometry

Proposition A.4 (Overlap divergence under representation)**.** *Let P_X^t and $P_X^{t'}$ denote covariate laws under two treatments and let $P_\phi^t, P_\phi^{t'}$ be their pushforwards under measurable ϕ. When $P_X^t \ll P_X^{t'}$,*

$$\chi^2(P_\phi^t \| P_\phi^{t'}) \le \chi^2(P_X^t \| P_X^{t'}),$$

and the gap equals the conditional variance of the original density ratio given $\phi(X)$ under $P_X^{t'}$.

Let $\rho_X = \mathrm{d}P_X^t / \mathrm{d}P_X^{t'}$. The pushforward density ratio obeys $\rho_\phi(\phi(X)) = \mathbb{E}_{P_X^{t'}}[\rho_X(X) \mid \phi(X)]$. Conditional Jensen therefore gives

$$\begin{aligned}\chi^2(P_\phi^t \| P_\phi^{t'}) &= \mathbb{E}_{P_X^{t'}}\big[\mathbb{E}(\rho_X \mid \phi(X))^2\big] - 1 \\ &\le \mathbb{E}_{P_X^{t'}}[\rho_X^2] - 1 = \chi^2(P_X^t \| P_X^{t'}).\end{aligned}$$

The law of total variance yields

$$\chi^2(P_X^t \| P_X^{t'}) - \chi^2(P_\phi^t \| P_\phi^{t'}) = \mathbb{E}_{P_X^{t'}}[\mathrm{Var}(\rho_X \mid \phi(X))].$$

This result concerns divergence after measurable projection. It does not order kernel locality or n_{eff} across arbitrary learned encoders; those depend additionally on metric scale, bandwidth, and the selected representation solution.

A.6 Proof of Lemma 4.2: Conditional Gaussian Mean Shift

Proof roadmap. The proof has four steps. Step 1 differentiates a Gaussian-smoothed joint density and identifies its outcome-score with a conditional kernel mean shift. Step 2 applies this identity to the data and model laws, turning attraction minus repulsion into a score mismatch. Step 3 proves that a zero score mismatch makes the two smoothed laws equal, and injectivity of the fixed smoothing operator then recovers the unsmoothed laws. Step 4 combines this generative fixed point with ODS, lossless decoding, and causal identification to obtain the interventional target.

Assumptions and notation. For a joint density R on context–outcome space, define the fixed linear smoothing operator $\mathcal{S}_h$ by

$$\overline{R}_h(c,z) = (\mathcal{S}_h R)(c,z) = \iint k_c(c,c')k_z(z,z')R(c',z')\,\mathrm{d}c'\,\mathrm{d}z'. \tag{18}$$

We assume differentiation may pass under the integral in Equation equation 18; this holds, for example, under the stated Gaussian outcome kernel and standard integrability conditions. For $p(c)$-almost every c, the smoothed densities are positive and continuously differentiable in z on a common connected domain Ω_c, and $q_\theta(z \mid c) > 0$ for almost every $z \in \Omega_c$. The operator $\mathcal{S}_h$ is injective on the class containing P and Q_θ. Finally, P and Q_θ have the same context marginal $p(c)$ as in Equation equation 8.

For the proof, let $k_z(z,z') = \exp(-\|z-z'\|^2/(2h_z^2))$ and define

$$M_R(c,z) = \frac{\iint k_c(c,c')k_z(z,z')(z'-z)R(c',z')\,\mathrm{d}c'\,\mathrm{d}z'}{\iint k_c(c,c')k_z(z,z')R(c',z')\,\mathrm{d}c'\,\mathrm{d}z'}.$$

Wherever $\overline{R}_h(c,z) > 0$,

$$M_R(c,z) = h_z^2 \nabla_z \log \overline{R}_h(c,z).$$

Proof of Lemma 4.2. The Gaussian derivative is

$$\nabla_z k_z(z,z') = \frac{z'-z}{h_z^2} k_z(z,z').$$

Differentiating Equation equation 18 under the integral therefore gives

$$\nabla_z \overline{R}_h(c,z) = \frac{1}{h_z^2} \iint k_c(c,c')k_z(z,z')(z'-z)R(c',z')\,\mathrm{d}c'\,\mathrm{d}z'.$$

Dividing by the positive denominator $\overline{R}_h(c,z)$ proves the claim. □

A.7 Proof of Lemma 4.3: Zero Score Mismatch

Under the assumptions above, suppose

$$\nabla_z \log \overline{P}_h(c,z) = \nabla_z \log \overline{Q}_{\theta,h}(c,z)$$

for $p(c)$-almost every c and almost every $z \in \Omega_c$.

Proof of Lemma 4.3. Fix any context c for which the assumptions hold. Positivity allows us to define

$$f_c(z) = \log \overline{P}_h(c,z) - \log \overline{Q}_{\theta,h}(c,z).$$

Its gradient is zero almost everywhere on the connected domain Ω_c. Hence f_c is constant there, so for some $\kappa(c) > 0$,

$$\overline{P}_h(c,z) = \kappa(c)\overline{Q}_{\theta,h}(c,z). \tag{19}$$

Let $C_z = \int k_z(z,z')\,\mathrm{d}z = (2\pi h_z^2)^{d_Z/2}$, which is independent of z'. Tonelli's theorem and the shared context marginal imply, for $R \in \{P, Q_\theta\}$,

$$\begin{aligned}\int \overline{R}_h(c,z)\,\mathrm{d}z &= C_z \iint k_c(c,c')R(c',z')\,\mathrm{d}c'\,\mathrm{d}z' \\ &= C_z \int k_c(c,c')p(c')\,\mathrm{d}c'.\end{aligned}$$

Thus the integrals over z are the same for P and Q_θ. Integrating Equation equation 19 yields $\kappa(c) = 1$, and hence $\mathcal{S}_h P = \mathcal{S}_h Q_\theta$. Injectivity of the fixed operator gives $P = Q_\theta$. Since both laws have marginal $p(c)$, disintegration yields $p(z \mid c) = q_\theta(z \mid c)$ almost everywhere. □

A.8 Proof of Theorem 4.4: Conditional Score Correspondence and Fixed Point

Proof of Theorem 4.4. Apply Lemma 4.2 first to $R = P$ and then to $R = Q_\theta$. Their difference is

$$\begin{aligned} V(c,z) &= M_P(c,z) - M_{Q_\theta}(c,z) \\ &= h_z^2 \left[\nabla_z \log \overline{P}_h(c,z) - \nabla_z \log \overline{Q}_{\theta,h}(c,z) \right], \end{aligned}$$

which proves Equation equation 14. The empirical attraction in Equation equation 9 estimates the first mean shift, while the empirical repulsion in Equation equation 10 estimates the second. Leave-one-out removes a particle's zero-displacement self-term from the model denominator.

If $V(c,z) = 0$, the displayed identity gives equality of the smoothed outcome-scores wherever the densities are positive. Positivity of $q_\theta(\cdot \mid c)$ on the common domain promotes the stated Q_θ-almost-everywhere equality to the domain required by Lemma 4.3. That lemma then yields $q_\theta(z \mid c) = p(z \mid c)$ almost everywhere, proving the fixed-point claim. □

A.9 Proof of Corollary 4.5: Decoded Causal Target

Proof of Corollary 4.5. Theorem 4.4 gives

$$q_\theta^Z(\cdot \mid \phi(x), t) = p^Z(\cdot \mid \phi(x), t).$$

Proposition A.1 transports this equality through the fixed decoder,

$$d_{\#} q_\theta^Z(\cdot \mid \phi(x), t) = p^Y(\cdot \mid \phi(x), t).$$

ODS and Proposition A.3 replace the representation-conditional law by $p^Y(\cdot \mid x, t)$, and Proposition 3.1 identifies the latter with $\mathcal{L}(Y(t) \mid X = x)$. Chaining these equalities proves the result. □

Why the kernel must remain fixed. The fixed-kernel qualifier makes the zero-field argument apply to one injective smoothing operator throughout optimization. Cross-fitting estimates ϕ and the kernel geometry on the complementary fold and then freezes them, so $\mathcal{S}_h$ remains fixed along every generator branch. This construction supplies the operator-level condition used by Lemma 4.3 and Theorem 4.4, while retaining the benefits of an outcome-predictive representation (Ober et al., 2021; Liu et al., 2020).

A.10 Proof of Proposition 4.1: Amortized Drift Optimization

Proof roadmap. We first condition on one Monte Carlo batch and regard its clipped field as a fixed vector. We then differentiate the detached regression target, map the resulting parameter step back to particle space through the generator Jacobian, and show nonnegative alignment with the prescribed transport. Finally, we characterize the gap between a parameter-space stationary point and the distributional zero field.

Notation and scope. Let $N = BS$ and stack the particles, clipped field vectors, and generator Jacobians as

$$\widehat{\mathbf{z}} = (\widehat{z}_1, \ldots, \widehat{z}_N), \qquad \mathbf{u} = (u_1, \ldots, u_N), \qquad \mathbf{J} = \nabla_\theta \widehat{\mathbf{z}}.$$

Here $u_l = \mathrm{clip}(V_l, v_{\max})$, where clipping is radial: $\mathrm{clip}(v, r) = v \min\{1, r/\|v\|_2\}$. The result characterizes one detached update and its population expectation; the stationary-point implication is obtained under the tangent-coverage condition stated in Corollary 4.6.

Proof of Proposition 4.1. During backpropagation, the numerical residual for particle l is

$$\widehat{z}_l - \mathrm{sg}[\widehat{z}_l + \eta u_l] = -\eta u_l,$$

but the term inside $\mathrm{sg}[\cdot]$ has zero derivative. The chain rule therefore gives

$$\nabla_\theta \mathcal{L}_{\mathrm{DriftCF}} = \frac{2}{N} \sum_l J_l^\top (-\eta u_l) = -\frac{2\eta}{N} \mathbf{J}^\top \mathbf{u},$$

which proves Equation equation 13. For a parameter learning rate α,

$$\theta^+ = \theta + \frac{2\alpha\eta}{N} \mathbf{J}^\top \mathbf{u}.$$

A first-order Taylor expansion of the generated particles gives

$$\widehat{\mathbf{z}}(\theta^+) - \widehat{\mathbf{z}}(\theta) = \frac{2\alpha\eta}{N}\mathbf{J}\mathbf{J}^\top\mathbf{u} + o(\alpha).$$

Because $\mathbf{K}_\theta = \mathbf{J}\mathbf{J}^\top$ is positive semidefinite,

$$\lim_{\alpha\downarrow 0}\frac{1}{\alpha}\left\langle \mathbf{u}, \widehat{\mathbf{z}}(\theta^+) - \widehat{\mathbf{z}}(\theta)\right\rangle = \frac{2\eta}{N}\|\mathbf{J}^\top\mathbf{u}\|_2^2 \geq 0.$$

Hence the infinitesimal update realizes the component of the desired particle motion available through the shared generator parameters.

At population level, write $\mathcal{J}$ for the linearized map from parameter perturbations to output perturbations and $\mathcal{J}^*$ for its adjoint. A zero semi-gradient gives $\mathcal{J}^*u = 0$. Under the tangent-coverage condition $u \in \overline{\mathrm{range}(\mathcal{J})}$, orthogonality to that range yields $\|u\| = 0$. Radial clipping satisfies $u = 0$ if and only if $V = 0$; hence zero population semi-gradient implies the zero field required by Theorem 4.4. □

Why this yields one-step generation. The particle calculation also induces a distributional evolution. Consider the continuous-time parameter flow corresponding to the population semi-gradient, $\dot\theta = 2\eta\,\mathbb{E}[J_\theta^\top u]$. For a particle $Z_\theta = g_\theta(E, C)$,

$$\dot Z_\theta = J_\theta(E, C)\dot\theta, \qquad w_\theta(c, z) = \mathbb{E}[\dot Z_\theta \mid C = c, Z_\theta = z].$$

For every smooth compactly supported test function f, the reparameterization formula gives the weak continuity equation

$$\frac{\mathrm{d}}{\mathrm{d}\tau}\mathbb{E}_{q_{\theta(\tau)}}[f(Z, C)] = \mathbb{E}_{q_{\theta(\tau)}}[\nabla_z f(Z, C)^\top w_\theta(C, Z)],$$

equivalently $\partial_\tau q_\theta + \nabla_z\cdot(q_\theta w_\theta) = 0$ in the distributional sense. The realized velocity w_θ is the ideal clipped drift filtered through shared generator tangents; with full tangent coverage it can represent that drift, while otherwise it follows only its realizable component.

The field is recomputed under the current pushforward after every training update, so parameter learning evolves q_θ through this continuity equation. The proposition establishes the population update direction, and Corollary 4.6 identifies its zero semi-gradient with the target fixed point under tangent coverage. Once these projected displacements have been compiled into g_θ, a fresh noise draw maps directly to a complete outcome vector.

A.11 Proof of the Ideal-Transport Lemma

For the proof of Lemma 4.7, fix a context c and suppose q_τ evolves by the continuity equation with

$$v_\tau(z, c) = \nabla_z \log p(z \mid c) - \nabla_z \log q_\tau(z \mid c).$$

Then

$$\frac{\mathrm{d}}{\mathrm{d}\tau}\,\mathrm{KL}(q_\tau(\cdot \mid c)\|p(\cdot \mid c)) = -\mathbb{E}_{q_\tau}\|v_\tau(Z, c)\|_2^2 \leq 0.$$

More generally, on the joint context–outcome space with fixed marginal $p(c)$, replacing v_τ by a self-adjoint positive-semidefinite tangent operator $\mathcal{K}_\theta v_\tau$ makes the $p(c)$-averaged conditional KL derivative $-\langle v_\tau, \mathcal{K}_\theta v_\tau\rangle_{L^2(p(c)q_\tau(z|c))} \leq 0$. A shared generator may couple contexts through $\mathcal{K}_\theta$, so this operator statement concerns the average rather than each context separately.

Proof of Lemma 4.7. Let $F(q) = \int q\log(q/p)\,\mathrm{d}z$. Using $\partial_\tau q = -\nabla_z\cdot(qv)$, conservation of mass, and integration by parts,

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}\tau}F(q_\tau) &= \int (1 + \log(q_\tau/p))\partial_\tau q_\tau\,\mathrm{d}z \\ &= \int q_\tau v_\tau^\top \nabla_z \log(q_\tau/p)\,\mathrm{d}z \\ &= -\int q_\tau\|v_\tau\|_2^2\,\mathrm{d}z.\end{aligned}$$

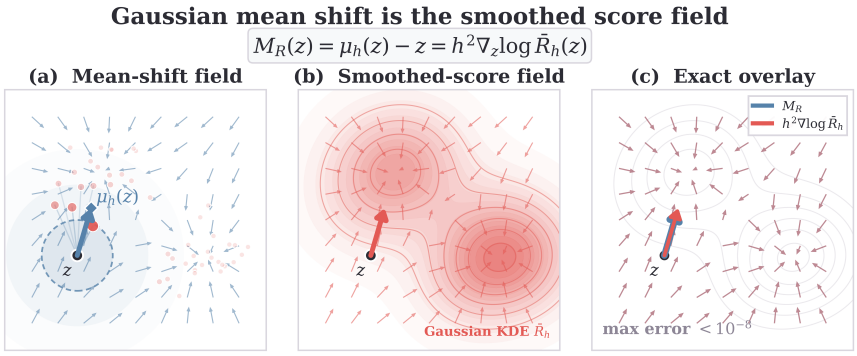


Figure 6: Gaussian mean shift and the corresponding scaled score of the smoothed density. The panels evaluate both constructions on the same locations and verify their pointwise agreement, providing the empirical identity used in Theorem 4.4.

The operator-valued statement follows by substituting $\mathcal{K}_\theta v_\tau$ for the velocity and using positive semidefiniteness. Integrating over $p(c)$ yields the same result for the conditional KL because DriftCF leaves the context marginal fixed. □

Lemma 4.7 describes the unsmoothed, unclipped ideal in which the drift equals the exact conditional score difference. The actual Gaussian-kernel field replaces both scores by scores of context–outcome smoothed joint densities, and clipping is nonlinear. For finite bandwidth and clipping, Theorem 4.4 supplies the appropriate fixed-point statement; Lemma 4.7 is motivation for the direction of evolution, not a monotonicity theorem for the implemented objective.

A.12 Proof of Corollary 4.6: Stationary Point to Causal Target

Proof of Corollary 4.6. By Proposition 4.1, the tangent-coverage condition converts a zero population semi-gradient into a zero clipped field and hence a zero unclipped field. Theorem 4.4 then gives equality of the branch's conditional adapted law and its target adapted law. Proposition A.5 transports this equality through the fixed decoder, ODS, and causal identification to obtain $\mathcal{L}(Y(t) \mid X = x)$. □

A.13 Proof of the Frozen-Pipeline Result

Proposition A.5 (Causal validity of the frozen cross-fitted pipeline)**.** *Condition on the nuisance fit from $\mathcal{I}_{-k}$. Suppose that the original data satisfy Proposition 3.1, $\phi^{(-k)}$ is a function only of pre-treatment X and satisfies ODS for an independent draw from the target population, $d^{(-k)} \circ a^{(-k)}$ is lossless, the queried context has support, and the assumptions of Theorem 4.4 hold for the resulting fixed kernel. Then a zero-field branch recovers $\mathcal{L}(Y(t) \mid X = x)$; an outcome-space mixture of valid branches has the same target law.*

Proof. Condition on the nuisance fit from $\mathcal{I}_{-k}$. It is then a collection of fixed measurable maps and constants, statistically independent of the outcomes used to form branch k's empirical field. Theorem 4.4 applies to this fixed branch geometry and gives

$$q^Z_{\theta_k}(\cdot \mid \phi^{(-k)}(x), t) = p^Z(\cdot \mid \phi^{(-k)}(x), t)$$

at a population zero field. Proposition A.1 transports this equality to the decoded outcome space. ODS in Proposition A.3 then replaces conditioning on $\phi^{(-k)}(x)$ by conditioning on x, and Proposition 3.1 identifies the resulting factual law with $\mathcal{L}(Y(t) \mid X = x)$. If every decoded branch measure equals this target, their outcome-space mixture also equals it by linearity of measures. □

B Gaussian-Kernel Drifting and Score-Based Mean Shift

Score-based models. Both score-based generation and drifting describe distributional evolution through score-like vector fields. A diffusion model introduces an artificial noise time τ, learns the score of every noised conditional law $p_\tau(z \mid c)$, and generates by integrating a reverse-time SDE or probability-flow ODE from a terminal noise distribution (Ho et al., 2020; Song et al., 2021). DriftCF introduces no diffusion time. At the current generator law, its population direction is the endpoint discrepancy

$$h_z^2 \big[\nabla_z \log \overline{P}_h(c, z) - \nabla_z \log \overline{Q}_{\theta,h}(c, z)\big].$$

The data score attracts particles, while subtracting the model score removes motion already explained by the current pushforward. Training repeatedly recomputes this discrepancy and changes Q_θ; inference evaluates the final noise-to-outcome map once. Thus the connection is at the level of score-driven distribution evolution, whereas the learned objects and inference procedures differ.

High-dimensional concentration. The Gaussian identity holds in arbitrary finite outcome dimension, and the score–drifting equivalence and its high-dimensional analysis are established by Lai et al. (2026). Pairwise Euclidean distances concentrate around a narrow shell as dimension grows; normalization and bandwidth selection set the shell that contributes to each mean shift. Within that shell, the weighted displacement remains exactly the gradient of the log-smoothed density by Equation equation 2. DriftCF applies this dimension-valid identity to complete outcome vectors and selects the smoothing scale from branch-external data.

Conditional flow matching. Conditional flow matching specifies a probability path $(p_\tau)_{\tau\in[0,1]}$, constructs target velocities from endpoint or conditional interpolation pairs, and trains a time-dependent field $v_\psi(\tau, z, c)$ whose ODE transports the base law to the data law (Lipman et al., 2023). Its statistical analyses connect the learned path to kernel density estimation and bound distributional error through velocity approximation (Kunkel & Trabs, 2025; Su et al., 2025). DriftCF instead recomputes an endpoint discrepancy between the data law and the generator's current pushforward, then projects it through the Jacobian of a time-independent generator. Flow matching integrates its learned velocity at sampling time; DriftCF compiles particle displacement into the generator parameters.

Causal conditioning. In all three paradigms, a context can be appended to a network input, but causal validity does not follow from conditional generation alone. DriftCF fixes the context marginal, restricts factual borrowing with k_T, requires the representation to preserve the conditional outcome law, and invokes Proposition 3.1 only on supported contexts. Consequently, its decoded equilibrium is an interventional law under the stated assumptions rather than merely an observationally conditioned generator. Neither diffusion, flow matching, nor drifting can recover hidden confounding or unsupported treatment regions by architecture alone.

C Theory-Aligned Encoder and Training Protocol

C.1 Outcome adaptation and treatment configuration

The nuisance stage uses an invertible conditional generative probe for the adapted outcome law. Its bijective noise-to-outcome map provides likelihood-free samples for the multivariate proper score while preserving the complete-vector dependence used to train ϕ. This probe is discarded after fitting; only the frozen covariate map enters DriftCF. The main generator need not be invertible because its role is to amortize the particle field rather than evaluate a likelihood.

The outcome adapter is fixed from nuisance-fold statistics. For a nonnegative continuous outcome,

$$z = \frac{\log(1+y) - \mu_{\log}}{\sigma_{\log}}, \qquad y = \exp(\sigma_{\log} z + \mu_{\log}) - 1.$$

For binary y, let $z = y + u$ with $u \sim \mathrm{Unif}(-\delta, \delta)$, $\delta < 1/2$, and decode $y = \mathbf{1}[z > 1/2]$. For count y, use $u \sim \mathrm{Unif}(-1/2, 1/2)$ and round to the valid count range. A zero-inflated continuous outcome uses a jittered occurrence coordinate and a transformed positive-magnitude coordinate; the

magnitude is ignored when the decoded occurrence indicator is zero. These augmented constructions retain enough information to decode the original mixed outcome exactly.

Treatment semantics and causal borrowing remain separate throughout training:

Treatment	$e_T(t)$	$k_T(t, t')$	Sharing rule
Binary	one-hot	exact match	within arm
Unordered	learned embedding	exact match	within arm
Ordinal	rank + embedding	Gaussian on rank	nearby levels
Continuous	cubic B-spline	Gaussian on dose	local doses

Design objective. The encoder has two roles and no third one: it supplies a low-dimensional conditioning variable and fixes the metric used to decide which factual units contribute to a local field. It does not identify hidden confounding, repair unsupported treatments, or generate outcomes. The protocol below isolates this geometry from the outcomes used to fit each generator branch and keeps it fixed throughout drift optimization.

We train the outcome-predictive encoder ϕ_ω with a lightweight conditional distributional probe q_ϑ and the multivariate energy score (Gneiting & Raftery, 2007):

$$\mathcal{L}_{\text{enc}}(\omega, \vartheta) = \frac{1}{|\mathcal{I}|} \sum_{i \in \mathcal{I}} S_{\text{ES}}(q_\vartheta(\cdot \mid \phi_\omega(x_i), e_T(t_i)), z_i) + \lambda \Omega(\omega), \tag{20}$$

where

$$S_{\text{ES}}(q, z) = \mathbb{E}_{Z' \sim q} \|Z' - z\|_2 - \frac{1}{2} \mathbb{E}_{Z', Z'' \sim q} \|Z' - Z''\|_2.$$

Strict propriety makes the population objective sensitive to the complete conditional distribution. Outcome-distribution sufficiency remains the explicit representation condition used by Proposition A.3; finite-sample minimization of Equation equation 20 is assessed through held-out distributional diagnostics.

The diagnostics compare distributional and mean probes from $\phi(X)$ and raw X, conditional covariance and rank dependence, treatment predictability, n_{eff}, singular values, pairwise distances, and cross-fold neighbor stability. These measurements distinguish useful nuisance removal from geometric collapse.

C.2 Cross-fitted branch construction

For branch k, every pairwise kernel evaluation uses one fold-external map, following cross-fitting's separation of nuisance estimation from target-stage fitting (Chernozhukov et al., 2018):

$$k_c(i, j) = k_x\big(\phi^{(-k)}(x_i), \phi^{(-k)}(x_j)\big) k_T(t_i, t_j), \qquad i, j \in \mathcal{I}_k. \tag{21}$$

At inference, complete decoded branch distributions are mixed in the original outcome space:

$$\widehat{q}^{\,Y}(\cdot \mid x, t) = \frac{1}{K} \sum_{k=1}^{K} d_{\#}^{(-k)} q_{\theta_k}^{Z}\big(\cdot \mid \phi^{(-k)}(x), t\big). \tag{22}$$

Proposition A.5 formalizes how the frozen nuisance components preserve the target across branches.

Before any nuisance fitting, randomly partition the available training sample into approximately equal folds $\mathcal{I}_1, \ldots, \mathcal{I}_K$ using a rule that does not depend on outcomes. Treatment-stratified splitting may be prespecified for discrete arms. For each k:

1. On $\mathcal{I}_{-k}$, fit all adapter statistics and covariate normalization. Train $(\phi^{(-k)}, q_\vartheta^{(-k)})$ with the multivariate proper score in Equation equation 20. Select representation dimension, regularization, and kernel bandwidths without using outcomes in $\mathcal{I}_k$.
2. Discard $q_\vartheta^{(-k)}$ and freeze the adapter, normalizers, $\phi^{(-k)}$, treatment kernel k_T, and bandwidths. Their computational graphs are detached from generator optimization. The generator-side treatment representation e_T remains a separate input path and may be optimized jointly with g_{θ_k} because it never changes k_c.

Algorithm 2 Cross-fitted DriftCF pipeline and distributional inference

Require: Observations $\mathcal{D} = \{(X_i, T_i, Y_i(T_i))\}_{i=1}^n$, folds $\{\mathcal{I}_k\}_{k=1}^K$, training particles S

for $k = 1, \dots, K$ **do**
  On $\mathcal{I}_{-k}$ fit X normalization, mixed-outcome adapter, $\phi^{(-k)}$ with its distributional probe, and kernel bandwidths.
  Freeze the nuisance geometry $\{a^{(-k)}, d^{(-k)}, \phi^{(-k)}, k_x, k_z, k_T\}$.
  On $\mathcal{I}_k$, compute $h_i = \phi^{(-k)}(X_i)$, $z_i = a^{(-k)}(Y_i, U_i)$, and causal context weights $k_c = k_x k_T$.
  repeat
    Draw $\epsilon_{is} \sim \mathcal{N}(0, I)$ and generate $\widehat{z}_{is} = g_{\theta_k}(\epsilon_{is}, h_i, e_T(T_i))$.
    Compute factual attraction A_{is} and leave-one-out model mean shift R_{is} under the same k_c.
    Set $V_{is} = A_{is} - R_{is}$ and $\widetilde{z}_{is} = \mathrm{sg}[\widehat{z}_{is} + \eta\, \mathrm{clip}(V_{is}, v_{\max})]$.
    Update only generator-side parameters with $\sum_{i,s} \|\widehat{z}_{is} - \widetilde{z}_{is}\|_2^2$.
  until the branch stopping criterion is met
end for
Query (x, t)**:** compute branch-wise $n_{\mathrm{eff}}(x, t)$ and local propensity support; reject or flag weakly supported queries.
for $s = 1, \dots, S_{\mathrm{eval}}$ **do**
  Draw branch $K_s \sim \mathrm{Unif}\{1, \dots, K\}$ and noise $\epsilon_s \sim \mathcal{N}(0, I)$.
  $\widehat{Y}^{(s)}(t) = d^{(-K_s)}\big(g_{\theta_{K_s}}(\epsilon_s, \phi^{(-K_s)}(x), e_T(t))\big)$.
end for
return complete-vector samples $\{\widehat{Y}^{(s)}(t)\}_{s=1}^{S_{\mathrm{eval}}}$ from the outcome-space branch mixture.

3. Transform every observation in $\mathcal{I}_k$ with this one frozen branch. All pairs in Equation equation 21 use $\phi^{(-k)}$ on both arguments; embeddings produced by different branches never enter the same Euclidean distance.
4. Fit g_{θ_k} on $\mathcal{I}_k$ using Equations equation 9–equation 12. Only θ_k is updated. Empty or weak neighborhoods are reported through Equation equation 11, not silently replaced by unsupported cross-treatment extrapolation.

For a new query (x, t), first sample K_0 uniformly from $\{1, \dots, K\}$, then sample noise and decode branch K_0. This sampling procedure has exactly the mixture law in Equation equation 22. It is valid for a new query because no query is assigned to a training fold. Averaging samples or distributional functionals in the original outcome space is also well defined; directly averaging parameters or latent coordinates is not, because independently trained representations are identifiable only up to transformations that need not preserve a shared Euclidean coordinate system.

Monte Carlo functionals. Each draw in Algorithm 2 uses a new ϵ_s and exactly one generator evaluation. Repeating this draw changes only Monte Carlo resolution, not the learned model or sampling path. For outcome coordinate m, the sample mean $S_{\mathrm{eval}}^{-1} \sum_s \widehat{Y}_m^{(s)}(t)$ estimates the conditional mean, empirical quantiles give prediction intervals, and the empirical covariance of the complete vectors estimates within-treatment dependence. For a measurable joint event $B \subseteq \mathcal{Y}$,

$$\widehat{\Pr}(Y(t) \in B \mid X = x) = \frac{1}{S_{\mathrm{eval}}} \sum_{s=1}^{S_{\mathrm{eval}}} \mathbf{1}\left[\widehat{Y}^{(s)}(t) \in B\right].$$

Thus means, quantiles, intervals, covariance, rank association, and joint risks all come from the same set of coherent vectors.

Although factual outcomes supervise $\phi^{(-k)}$ on $\mathcal{I}_{-k}$, the deployed encoder is a fixed map of pre-treatment covariates alone. For an independent query, neither its factual treatment nor its factual outcome enters $\phi^{(-k)}(x)$; treatment is supplied only as the requested condition. Cross-fitting strengthens this separation during branch training by ensuring that the outcomes used to construct branch k's field did not fit that branch's neighborhood map. The outer test split remains untouched by adapter fitting, representation learning, bandwidth selection, generator fitting, and early stopping.

C.3 What is preserved, and what remains assumed

The implementation follows a compact contract with the population analysis:

1. **Pre-treatment context:** ϕ receives only measured pre-treatment X; the requested treatment enters separately through e_T and k_T.
2. **Fixed conditional geometry:** normalization, ϕ, k_T, and all bandwidths are fitted on $\mathcal{I}_{-k}$ and frozen before fitting branch k.
3. **Outcome-law preservation:** the proper-score probe targets ODS, and the fixed adapter satisfies $d \circ a = \mathrm{id}$ on observed outcomes.
4. **Shared context marginal:** DriftCF generates only Z conditional on an observed or queried context; it does not generate or reweight C.
5. **Supported estimation:** exact treatment masks or local treatment kernels define borrowing, while n_{eff} and local propensity diagnostics record available support.
6. **Correct drift equilibrium:** factual attraction and leave-one-out model repulsion are computed under the same frozen kernel, with no term that changes the target conditional law.

These mechanisms preserve the conditions invoked by Propositions 3.1 and A.5; the corresponding empirical diagnostics are specified in Appendix E.

C.4 Finite-sample approximations and diagnostics

Several implementation devices determine the finite-sample field. Leave-one-out repulsion removes the deterministic zero-displacement self-term that would otherwise inflate the empirical model denominator. The stabilizer ε_0 prevents numerical division by zero, radial clipping preserves field direction and zeros while controlling magnitude, and chunked pairwise computation accumulates exact numerators and denominators up to floating-point arithmetic. Bandwidths and clipping thresholds are selected without test outcomes and frozen during final generator fitting.

The diagnostic protocol compares distributional and mean probes from $\phi^{(-k)}(X)$ and raw X, conditional covariance and rank dependence, treatment predictability, n_{eff}, singular values, pairwise distances, and cross-fold neighbor stability. During generator training, Jacobian–vector and vector–Jacobian products evaluate alignment between $\mathbf{K}_\theta \mathbf{u}$ and $\mathbf{u}$ without materializing $\mathbf{K}_\theta$. Together these checks connect representation geometry, local support, and tangent coverage to the assumptions used in the population analysis.

D Support, Implementation, and Reporting Details

D.1 Support protocol

For every query, we compute Equation equation 11 and, for continuous treatment, local generalized propensity support. Metrics are reported over all supported queries and a prespecified adequate-support subset; the coverage–accuracy curve records the effect of varying the support threshold. Dose-wise n_{eff} exposes boundary regions where local smoothing is unreliable.

D.2 Default architecture and optimization

The default protocol uses $K = 3$ complete cross-fitted branches as specified in Appendix C. Within branch k, nuisance components are fitted on $\mathcal{I}_{-k}$ and g_{θ_k} is fitted on $\mathcal{I}_k$; predictions are combined only as decoded outcome-space samples through Equation equation 22. The default encoder is an MLP $X \to 256 \to 128 \to d_\phi$ with SiLU activations. The probe is a two-layer conditional generator of width 128. The main generator consumes $[\epsilon, \phi(X), e_T(T)]$ and uses widths $256, 256, 128$ before outputting d_Z coordinates. The branch configuration uses $d_\phi = 32$, $d_T = 8$, $d_\epsilon = 16$, $S = 4$ training particles, and $S_{\mathrm{eval}} = 100$ evaluation particles, selected on branch-external validation data.

Gaussian k_x and k_z bandwidths are initialized by a median heuristic and selected using only branch-external data. Ordinal and continuous treatment-kernel bandwidths are selected likewise and frozen before generator fitting. A learned treatment embedding belongs to the generator conditioning path and does not enter k_T. Pairwise context and outcome kernels are computed in chunks; leave-one-out repulsion masks the diagonal rather than materializing deleted arrays. The step size η, clipping

threshold $v_{\max}$, and stabilizer ε_0 are selected without access to generator-fold or test outcomes. The population theorem uses exact normalized expectations with $\varepsilon_0 = 0$; the finite-sample implementation tracks neighborhood denominators and n_{eff} alongside the stabilized field.

D.3 Reproducibility record

For each benchmark, the experimental record fixes the dataset version, treatment and outcome definitions, train/validation/test indices, fold count, random seeds, evaluation draw count, support threshold, baseline adaptation status, and hardware. Semi-synthetic benchmarks additionally fix the potential-outcome generator used for ground-truth evaluation.

E Supplementary Experiments

E.1 Benchmark alignment and baseline implementation

The binary-treatment experiments follow DIME's semi-synthetic protocol for learning a multi-outcome joint interventional distribution and evaluating empirical Wasserstein-1 distance within and across treatment arms on ACIC and MIMIC-III (Ma et al., 2025). We inherit DIME's data construction, preprocessing, outcome mechanisms, and ten-fold evaluation protocol, then apply the same splits, evaluation draws, and outcome normalization to every method. Thus the MIMIC results use real high-dimensional clinical covariates and treatment labels with DIME's semi-synthetic mixed-outcome potential outcomes, which provide known treatment-specific joint distributions for W_1 evaluation.

DIME is native for binary treatment; we extend its treatment condition and mask to multi-valued, ordinal, and continuous inputs for DIME*. DiffPO and the intervention-density normalizing flow are fitted outcome-wise and combined as marginal products. PO-Flow is evaluated only on the shared target $p(Y(t) \mid X = x)$; its factual-outcome-conditioned cross-world task is excluded. GANITE, CEVAE, SCIGAN, S-/T-Net, TARNet, and CFRNet use repeated stochastic forward passes or MC dropout, followed by a marginal-product joint. CCME and kernel conditional density estimation are restricted to low-dimensional settings. Every adaptation is marked *, and native plus adapted results are separated in the detailed tables.

E.2 Datasets and additional treatment settings

The complete benchmark suite comprises:

- **ACIC:** binary treatment with real covariates and a known mixed-outcome potential-outcome mechanism, used for DIME-aligned W_1 evaluation.
- **IHDP:** real covariates and treatment assignment with heterogeneous mixed outcomes, used to test conditional joint-law recovery.
- **News:** continuous treatment with a known dose-dependent multi-outcome mechanism, used to test local treatment geometry.
- **MIMIC-III:** DIME's semi-synthetic benchmark built from high-dimensional clinical covariates and mechanical-ventilation treatment labels, with known mixed length-of-stay and mortality-related potential outcomes for treatment-specific joint-distribution evaluation.
- **Industrial deployment:** continuous incentive intensity with *(cost,watching time, DAU, revenue)* as the mixed outcome vector. Industrial figures use held-out real observations.

E.3 Complete benchmark tables

The detailed counterpart of Table 1 reports empirical Wasserstein-1 distances $\widehat{W}_1^{\text{in}}$ and $\widehat{W}_1^{\text{out}}$ separately for each treatment arm on the ACIC and MIMIC semi-synthetic benchmarks. Table 7 reports multivariate energy distance and dependence error for IHDP and News. All mixed-outcome metrics are computed after decoding complete outcome vectors.

Table 5: Computation under matched hardware and batch size. Inference statistics use complete joint draws; peak memory is normalized to DriftCF.

Method	NFE	Median latency	p95 latency	peak memory
DIME	40	4.71 ms	5.22 ms	1.26×
Joint conditional diffusion	115	4.81 ms	5.29 ms	1.18×
PO-Flow	32	3.64 ms	4.89 ms	1.08×
Joint conditional flow matching	40	3.22 ms	4.48 ms	1.23×
DriftCF	1	2.82 ms	4.63 ms	1.00×

E.4 Controlled-objective and field ablations

The controlled comparisons in this subsection use the same initialization, generated particles, factual minibatch, context kernel, outcome kernel, adapter, generator architecture, batch construction, and compute budget. They differ only in the training objective or field component:

- **Data mean shift:** M_P, equivalent to attraction without model correction.
- **DriftCF:** $M_P - M_Q$, the normalized data–model mean-shift difference.
- **Conditional MMD gradient:** the particle gradient of a conditional characteristic-kernel MMD objective, rather than the value of the scalar loss alone.

For a generated particle $\widehat{z}_l$ and normalized context weights w_{lj}, data mean shift uses the ratio-normalized displacement

$$u_l^{\mathrm{MS}} = \frac{\sum_j w_{lj} k_z(\widehat{z}_l, z_j)(z_j - \widehat{z}_l)}{\sum_j w_{lj} k_z(\widehat{z}_l, z_j)},$$

whereas DriftCF subtracts the analogously normalized model mean shift. For the MMD comparison, define the conditional empirical discrepancy

$$\widehat{\mathrm{MMD}}_c^2 = \sum_{j,j'} \alpha_{jj'} k_z(z_j, z_{j'}) + \sum_{l,l'} \beta_{ll'} k_z(\widehat{z}_l, \widehat{z}_{l'}) - 2 \sum_{l,j} \gamma_{lj} k_z(\widehat{z}_l, z_j),$$

where α, β, γ are normalized products of the same context weights used by DriftCF. Its particle direction is $u_l^{\mathrm{MMD}} = -\nabla_{\widehat{z}_l} \widehat{\mathrm{MMD}}_c^2$. Unlike the ratio-normalized mean shifts, this direction inherits the global weighting of the scalar discrepancy. Conditional flow matching learns $v_\psi(\tau, z, c)$ against the velocity of a specified interpolation Z_τ and obtains a sample by integrating $\dot{Z}_\tau = v_\psi(\tau, Z_\tau, c)$. For the endpoint directions, Table 2 compares cosine and MSE against the known population drift at the same particle locations, followed by the distribution reached under an equal training-compute budget. Conditional flow matching is evaluated separately as a time-indexed path method: it uses the same endpoints and architecture budget, but only final-distribution and sampling-cost metrics are compared because its interpolation velocity is not the endpoint score mismatch.

E.5 Particle count, batch size, and effective support

We evaluate training particles $S \in \{1, 2, 4, 8, 16\}$ jointly with factual batch size and query support because repulsion uses approximately $BS - 1$ model neighbors while attraction uses factual neighbors selected by k_c. Figure 7 summarizes joint-distribution error over (B, S), agreement with the population field across n_{eff}, the coverage–accuracy trade-off induced by support thresholds, and attraction and repulsion magnitudes across support bins.

E.6 Computation and amortization

All inference measurements use the same hardware, query batch, outcome dimension, evaluation draw count, and GPU warm-up. Diffusion models retain their sampling steps, flow models retain their solver tolerances and realized NFE, and DIME retains its outcome ordering. Table 5 reports NFE, median and p95 latency, and peak memory; Figure 5 shows the corresponding quality–cost frontier.

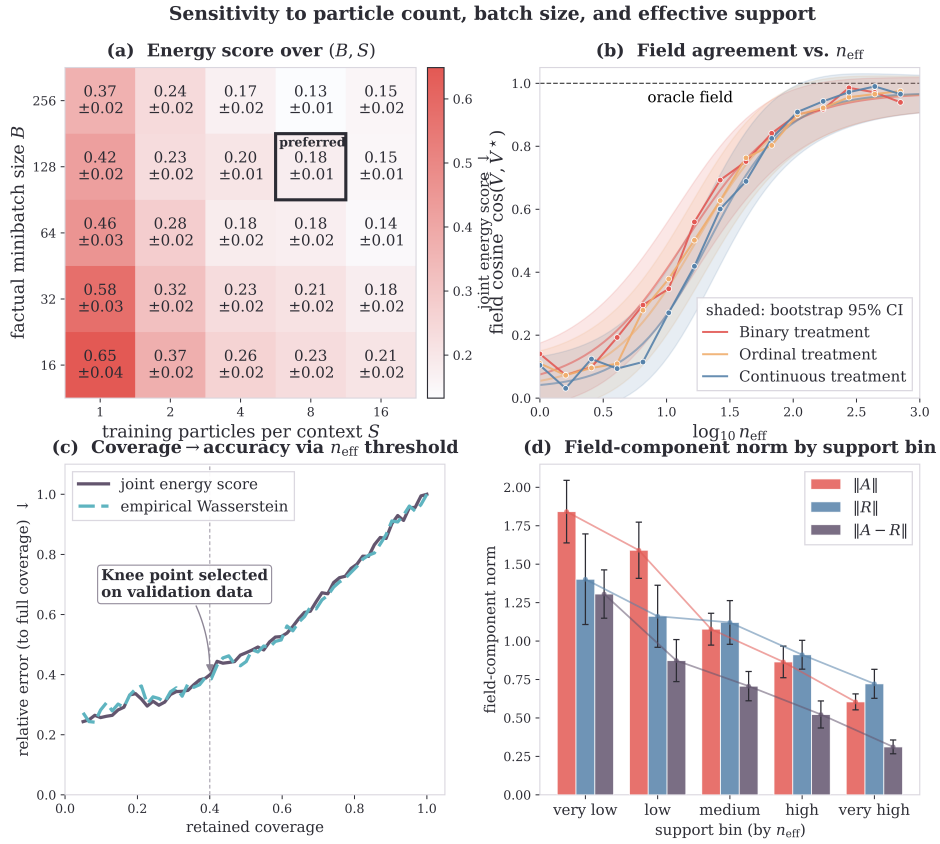


Figure 7: Sensitivity to particle count, factual batch size, and effective support. The four panels summarize joint-distribution error over (B, S), agreement with the population field across n_{eff}, the coverage–accuracy trade-off induced by support thresholds, and the magnitudes of attraction and repulsion across support bins.

E.7 Visual evidence for drifting

Figures 3, 6, and 7 provide complementary views of the drifting mechanism: conditional attraction–repulsion, pointwise agreement between Gaussian mean shift and the scaled smoothed score, and sensitivity to particle count and effective support. Figure 5 complements these mechanism views with quality versus inference cost.

E.8 Derived functionals and observed-cohort reporting

For every supported query, Algorithm 2 generates complete vectors from which means, quantiles, intervals, covariance, rank association, and joint-event probabilities are computed without retraining. The benchmark tables evaluate these distributions only where the semi-synthetic potential-outcome mechanism provides the corresponding ground truth; the industrial study evaluates held-out joint outcomes, serving cost, and randomized policy impact.

E.9 Online incentive-allocation protocol

The online study uses user-level randomization with a stable assignment key and three arms: the point-estimate policy, a point-estimate policy, and the DriftCF distributional policy. All arms share the eligible population, candidate incentive set, budget accounting, serving constraints, and prede-clared exclusion rules. DriftCF allocates continuous incentive intensity using joint samples of cost, watching time, DAU, and revenue.

Table 6: Detailed binary-treatment results on the ACIC and MIMIC semi-synthetic benchmarks. Each cell reports mean ± standard deviation over ten folds; W_1 is the empirical Wasserstein-1 distance.

Dataset	Method	$W_{1,0}^{\text{in}} \downarrow$	$W_{1,0}^{\text{out}} \downarrow$	$W_{1,1}^{\text{in}} \downarrow$	$W_{1,1}^{\text{out}} \downarrow$
ACIC	DIME	1.76(0.22)	1.85(0.15)	0.57(0.12)	1.02(0.09)
	DiffPO product*	1.84(0.12)	2.24(0.16)	1.14(0.09)	1.26(0.14)
	PO-Flow*	1.58(0.14)	1.74(0.10)	0.44(0.05)	0.95(0.15)
	Joint diffusion	2.44(0.29)	3.05(0.27)	1.21(0.20)	2.25(0.32)
	Joint flow matching	1.86(0.09)	2.65(0.14)	0.52(0.11)	1.45(0.17)
	DriftCF	1.34(0.11)	1.63(0.08)	0.37(0.06)	0.87(0.12)
MIMIC	DIME	3.53(0.19)	3.86(0.24)	1.24(0.12)	1.64(0.11)
	DiffPO product*	3.72(0.20)	4.24(0.42)	1.75(0.17)	1.92(0.17)
	PO-Flow*	3.43(0.27)	3.71(0.36)	1.65(0.19)	1.77(0.23)
	Joint diffusion	3.14(0.16)	3.42(0.33)	1.98(0.16)	2.34(0.31)
	Joint flow matching	3.17(0.12)	3.65(0.29)	1.66(0.24)	2.08(0.22)
	DriftCF	2.98(0.14)	3.17(0.25)	1.08(0.15)	1.29(0.14)

Table 7: Detailed IHDP and News results. Energy and dependence errors are computed on held-out complete outcome vectors. * denotes an adaptation.

Method	IHDP Energy ↓	IHDP Depend. ↓	News Energy ↓	News Depend. ↓
DIME*	0.82(0.06)	0.16	3.12(0.23)	0.14
DiffPO product*	0.89(0.12)	0.17	2.71(0.36)	0.15
PO-Flow*	0.44(0.04)	0.15	2.65(0.25)	0.12
Joint diffusion	0.77(0.09)	0.16	2.54(0.14)	0.13
Joint flow matching	0.59(0.13)	0.14	2.77(0.08)	0.13
DriftCF	0.29(0.05)	0.08	2.36(0.18)	0.06

The primary analysis uses intention-to-treat relative lifts in DAU, watching time, video view count, and ad revenue. Table 4 presents the predeclared DriftCF-versus-production comparison with randomization-unit-level p-values. Offline model selection is completed before randomization, and no online outcome is used to retune a running arm.

E.10 Evaluation FAQ

Why is DIME the primary joint-distribution baseline? DIME natively targets the same binary-treatment multi-outcome interventional law. Its autoregressive diffusion sampler also provides the direct contrast to simultaneous one-step generation. Extensions beyond its published setting are reported separately as adaptations.

Why is PO-Flow not evaluated on factual-conditioned counterfactual recovery? That task conditions on the realized factual outcome and uses additional cross-world structure. DriftCF targets the within-treatment law identified by Equation equation 5; only the potential-outcome distribution task is shared.

Why repeat samples if generation is one step? One step refers to the neural path for a single draw. Distributional summaries require independent draws ϵ_s, each of which still uses one generator evaluation. Increasing S_{eval} reduces Monte Carlo error without introducing a denoising or integration trajectory.

How are weak-support queries handled? They remain visible through n_{eff} and local propensity diagnostics. Figure 7 shows the coverage–accuracy effect of varying the support threshold.

F Discriminative and Balancing Representations

Outcome-predictive and balancing representations optimize different objectives (Johansson et al., 2016; Shalit et al., 2017; Hansen, 2008). The former seeks a projection that retains the conditional outcome law; the latter suppresses information predictive of treatment assignment. For a

distribution-valued estimand, either objective can fail in finite samples: an outcome probe can underfit and discard distributional structure, while an aggressive balancing penalty can remove a confounder or effect modifier, a failure mode consistent with representation-induced bias analyses (Melnychuk et al., 2024b). Kernel geometry adds a third axis. A representation may increase n_{eff} by useful removal of nuisance dimensions or by harmful distance collapse. Consequently, no single observed quantity establishes representation quality.

DriftCF does not include balancing as an unqualified default because factual validation can reward large neighborhoods without detecting the mixture bias in Equation equation 17. Instead, the main model prioritizes outcome-distribution preservation, while the balancing penalty is an ablation evaluated jointly on distributional probes, locality, n_{eff}, and counterfactual metrics where oracle outcomes exist.

Sufficiency is invariant to invertible transformations of $\phi(X)$, whereas an isotropic Gaussian kernel is invariant only to orthogonal transformations and can absorb global scaling through bandwidth. More general invertible maps alter Euclidean neighborhoods. The practical method therefore freezes normalization and branch-external bandwidth selection together with the learned representation, preserving one kernel geometry throughout each generator branch.