ADAPTING CAUSAL EFFECT FOUNDATION MODEL FROM BINARY TO CONTINUOUS TREATMENTS

Anonymous authors
Paper under double-blind review

ABSTRACT

Causal effects describe how outcomes differ under alternative interventions and play an important role in medicine, economics, and public policy. Accurate causal effect estimation is therefore essential for informed decision-making. Recently, several studies have introduced causal effect foundation models based on in-context learning, achieving promising performance in causal effect estimation on new tasks without updating the parameters of the foundation model. However, existing causal effect estimation foundation models are primarily designed for binary treatments and cannot be directly applied to continuous treatment settings. In this paper, to leverage the strong predictive capability of causal effect foundation models for binary treatments in continuous treatment settings, we propose a lightweight trainable plugin to adapt a frozen causal effect foundation model of binary treatments to continuous treatment effect estimation. Specifically, we use treatment quantiles as thresholds to binarize the continuous treatment, yielding a series of anchor treatment values. For each anchor, we learn a covariate–treatment interaction representation through a task-specific encoder and feed it into a binary-treatment foundation model, whose parameters are kept fixed while the encoder is fine-tuned. Since each anchor model is expected to provide more accurate counterfactual estimates for target treatments close to its corresponding anchor value, we aggregate predictions from multiple anchor models to obtain the final prediction, with larger weights assigned to anchors closer to the target treatment. We conduct experiments on synthetic and semi-synthetic benchmark datasets to demonstrate the effectiveness of our method for continuous treatment effect estimation.

1 INTRODUCTION

Causal effect estimation studies how outcomes change under interventions and supports decisions in medicine, economics, and public policy (Rubin, 1974; Imbens & Rubin, 2015; Pearl, 2009). In the binary setting, it compares potential outcomes under two treatment states, such as receiving or not receiving an intervention (Rosenbaum & Rubin, 1983; Hill, 2011). Nevertheless, many practical decisions also require choosing the intensity of an intervention, which calls for a continuous treatment formulation (Imai & Van Dyk, 2004). Continuous treatment effect estimation characterizes how outcomes would change across treatment levels (Hirano & Imbens, 2004). Such estimates help decision makers choose an appropriate intervention level and allocate resources according to its expected effects (Schwab et al., 2020; Fong et al., 2018).

In recent years, in-context learning has emerged as a new learning paradigm in which a pretrained model makes predictions for a new task from examples supplied in its input, without updating its parameters (Brown et al., 2020). Prior fitted networks develop this paradigm by pretraining across simulated tasks, and use labeled observations as context for predictions on new datasets (Müller et al., 2021; Hollmann et al., 2022). Recent causal foundation models specialize in this paradigm by learning from simulated causal tasks to estimate treatment effects from observational context. Models such as DoPFN and CausalPFN achieve strong performance on binary treatment effect estimation tasks and outperform conventional causal estimators in benchmark comparisons (Robertson et al., 2026; Balazadeh Meresht et al., 2026).

However, the above causal effect foundation models are primarily designed for binary treatment interfaces, which cannot directly accept the continuous treatment variable required by continuous

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.

treatment effect estimation tasks. Reducing this variable to a binary label alone discards differences between continuous treatment values within both the treated and control groups. The central challenge is therefore to construct binary inputs that enable causal effect foundation models for binary treatments to be effectively leveraged, while preserving the continuous information needed to estimate the response function. To address this, we seek to reuse the causal task priors that these foundation models learn during pretraining and leverage their ability to generalize to continuous treatment effect estimation.

Specifically, we propose a Treatment Aware Plugin for Causal Effect Foundation Models (TAP-CEFM). The plugin uses treatment quantiles as anchors to construct local binary classification tasks. At each anchor, the treatment is locally binarized into binary labels, thereby defining the corresponding control and treated groups. To avoid losing continuous treatment information during this local binarization, we learn representations using the covariates, the original continuous treatment, and the binarized treatment labels. The learned representations are then fed into the frozen pretrained model for prediction, and the prediction errors produced by the frozen model are used to train the plugin. Relying on a single anchor prediction would make the accuracy of the final estimate depend entirely on that model. We therefore adopt weighted aggregation to reduce this dependence and sensitivity to errors from individual anchor models. Because nearby anchor models are expected to predict more accurately, we use a Gaussian function to weight anchors according to their distance from the target treatment. This assigns greater weight to nearby anchors, and Gaussian aggregation is finally employed to integrate the predictions across different anchors. We use two distinct causal effect foundation models as primary predictors to evaluate the plugin's generalizability across causal foundation models.

Our contributions are threefold. First, we design an input plugin that converts continuous treatment observations into local binary tasks for a frozen causal effect foundation model for binary treatments. Second, we preserve continuous treatment information through learned covariate interactions and combine local predictions with Gaussian aggregation. Third, we evaluate the approach on Synthetic, IHDP, and News, and study the plugin's components and parameter sensitivity.

2 Related Work

Causal effect estimation studies how interventions change outcomes rather than merely predicting outcomes from observed associations (Rubin, 1974; Rosenbaum & Rubin, 1983). Confounding arises when common causes influence both treatment assignment and outcomes (Pearl, 2009). For example, severe illness may lead to more intensive treatment and poorer recovery, which makes direct outcome comparisons mix treatment effects with initial health differences. Binary estimation compares two intervention states to estimate average effects and their heterogeneity (Künzel et al., 2019; Wager & Athey, 2018). Meta learners and causal forests model effect heterogeneity (Künzel et al., 2019; Wager & Athey, 2018), while neural methods address treatment imbalance through balanced representations or joint outcome and propensity modeling (Shalit et al., 2017; Shi et al., 2019). These methods fit estimators on each target dataset.

Continuous estimation instead recovers a response curve over treatment levels (Hirano & Imbens, 2004; Nie et al., 2021). Generalized propensity scores adjust for continuous treatment assignment (Hirano & Imbens, 2004), while kernel estimators combine local information sharing with doubly robust adjustment (Kennedy et al., 2017). Neural methods further address response variation. DRNet specializes prediction heads to treatment intervals (Schwab et al., 2020), but switching heads can introduce discontinuities. VCNet instead varies network coefficients with treatment to preserve continuity (Nie et al., 2021). ORIC uses optimal transport to align conditional covariate distributions with the marginal distribution (Yan et al., 2025). These conventional estimators fit models on each target dataset without reusing a pretrained causal predictor through context. Our work instead adapts a frozen predictor through an input plugin that constructs local binary tasks. Gaussian aggregation combines the resulting responses across treatment levels.

Prior Fitted Models and Prediction from Context. Representation pretraining transfers features across tables, as in TransTab and XTab (Wang & Sun, 2022; Zhu et al., 2023). In-context learning allows pretrained models to predict from contextual examples without parameter updates (Brown et al., 2020). Within this paradigm, prior fitted networks pretrain over task distributions to approximate Bayesian prediction from context (Müller et al., 2021). Tabular foundation models such as TabPFN

and TabICL apply this paradigm to prediction (Hollmann et al., 2025; Qu et al., 2025). Their training objectives target observed labels or responses. Causal effect estimation foundation models such as CausalPFN and DoPFN instead use priors, training objectives, and predictive interfaces that address causal quantities (Balazadeh Meresht et al., 2026; Robertson et al., 2026). CausalPFN trains on identifiable causal tasks using expected potential outcomes as supervision, while DoPFN learns to predict interventional outcomes from simulated causal structures. Both models report strong performance on binary treatment benchmarks (Balazadeh Meresht et al., 2026; Robertson et al., 2026). By reusing pretrained causal knowledge, they provide an alternative to fitting a new predictor for each dataset. This task alignment motivates our use of a causal foundation model as the frozen predictor.

Task Adaptation. Adaptation methods modify the context, pretrained parameters, or input interface. TuneTables optimizes a compact learned context and can also fine tune the predictor (Feuer et al., 2024). LoCalPFN combines retrieval of relevant observations with downstream fine tuning (Thomas et al., 2024). TabPFN Unleashed instead learns lightweight input encoders while keeping the pretrained predictor frozen (Liu & Ye, 2025). This strategy adapts the representation without updating the backbone. We extend this input adaptation idea and develop a treatment aware plugin that connects continuous causal estimation to a frozen binary causal predictor.

3 METHOD

We propose TAP-CEFM, an input plugin that adapts a frozen causal effect foundation model designed for binary treatments to continuous treatment estimation. The method first constructs supported binary tasks around treatment anchors. It then learns a representation that preserves continuous treatment information and covariate interactions. Each task uses the frozen predictor to estimate outcomes from context. After that, adaptive Gaussian aggregation combines these local predictions into a response function. Figure 1 presents the overall flow.

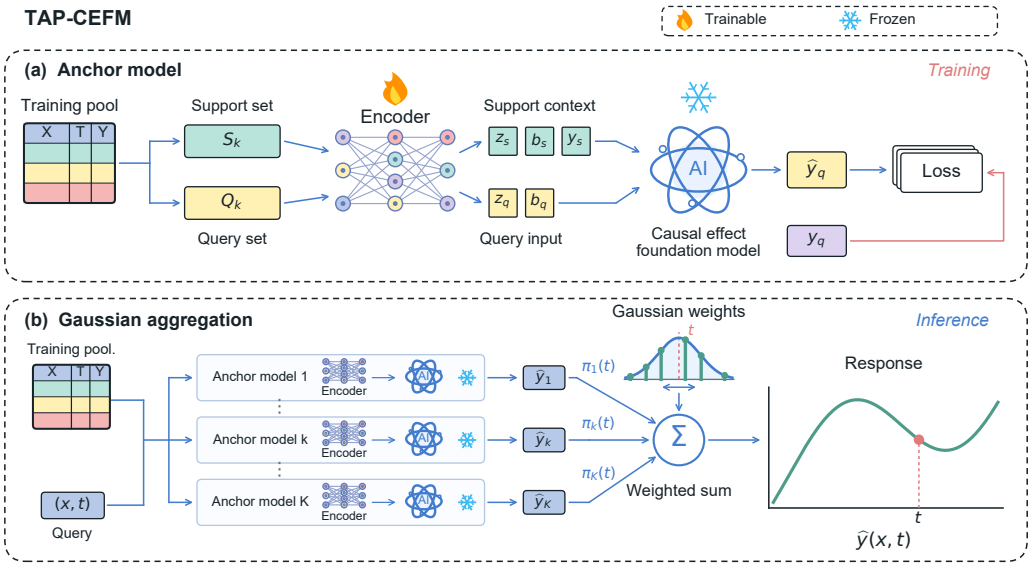


Figure 1: Overview of TAP-CEFM. Panel (a) constructs local binary tasks and trains an input plugin through frozen CausalPFN. Flames indicate trainable plugins and snowflakes indicate the frozen predictor. Panel (b) uses the complete training context and aggregates expert predictions to estimate the ADRF.

3.1 ANCHOR MODEL CONSTRUCTION

Causal effect foundation models use in-context learning to predict potential outcomes without updating their pretrained parameters (Appendix A.2).To adapt causal effect foundation models with

binary treatment interfaces to continuous treatment effect estimation, we design an anchor model that combines a trainable encoder with a frozen causal effect foundation model. The encoder uses representation learning to adapt continuous treatment data to the binary interface before the foundation model predicts outcomes. We next analyze factors governing the anchor model's local prediction error bound to explain our design. For covariates $x \in \mathbb{R}^p$ and treatment $t \in [0,1]$, the target is $\mu^\star(x,t) = \mathbb{E}[Y(t) \mid X = x]$, where $Y(t)$ is the potential outcome under treatment t. Anchor $\tau_k \in [0,1]$ defines the binary label

$$b_k(t) = \mathbb{I}\{t \geq \tau_k\}. \tag{1}$$

Here k indexes the anchor, and $b_k(t)$ equals zero when $t < \tau_k$ and one otherwise. Accordingly, let $\widehat{y}_{k,b}(x,t)$ denote the prediction with binary input $b \in \{0,1\}$, and write the actual prediction as $\widehat{y}_k(x,t) = \widehat{y}_{k,b_k(t)}(x,t)$. The following analysis examines the error of trained anchor models across different treatment values.

First, we suppose that the true response satisfies

$$|\mu^\star(x,u) - \mu^\star(x,v)| \leq L_\mu(x)|u-v|, \qquad u,v \in [0,1]. \tag{2}$$

For a given anchor k and treatment t, the causal effect foundation model receives binary input $b = b_k(t)$. We also suppose that its prediction function satisfies

$$|\widehat{y}_{k,b}(x,u) - \widehat{y}_{k,b}(x,v)| \leq L_{k,b}(x)|u-v|, \qquad u,v \in J_{k,b}. \tag{3}$$

Here u and v are treatment values. The intervals $J_{k,0} = [0,\tau_k]$ and $J_{k,1} = [\tau_k,1]$ specify the two sides of the anchor. Define the reference error for the selected binary label as

$$\varepsilon_k(x;t) = \begin{cases} |\mu^\star(x,\tau_k) - \widehat{y}_{k,0}(x,\tau_k)|, & t < \tau_k, \\ |\mu^\star(x,\tau_k) - \widehat{y}_{k,1}(x,\tau_k)|, & t \geq \tau_k. \end{cases} \tag{4}$$

This error evaluates the selected label's prediction at the anchor.

To guide the design of the anchor model, we introduce Theorem 1, which bounds prediction error using reference accuracy and treatment distance.

Theorem 1 (Local prediction error)**.** *Under the regularity conditions in Equations 2 and 3, the local prediction error satisfies*

$$|\mu^\star(x,t) - \widehat{y}_k(x,t)| \leq \varepsilon_k(x;t) + \big(L_\mu(x) + L_{k,b_k(t)}(x)\big)|t-\tau_k|. \tag{5}$$

In Equation 5, the term $\big(L_\mu(x) + L_{k,b_k(t)}(x)\big)|t-\tau_k|$ bounds the combined absolute changes in the true response and the selected model prediction between anchor τ_k and treatment t. For anchor k and treatment t, reducing $\varepsilon_k(x;t)$ decreases the reference error contribution to this bound. Using continuous treatment information to improve prediction accuracy at the anchor can reduce this reference error. We therefore design an encoder that preserves this information and learns to use a causal effect foundation model as its frozen predictor.

Continuous treatment encoder. An anchor assigns the same binary label to different treatment values on the same side of its threshold. To preserve these distinctions, the Treatment Adapter $\mathcal{T}_k$ embeds the treatment, the anchor and their signed difference

$$e_k(t) = \mathcal{T}_k(t, \tau_k, t-\tau_k). \tag{6}$$

The embedding $e_k(t) \in \mathbb{R}^{d_t}$ contains treatment information and its position relative to the anchor. Because treatment responses also depend on the covariates, the Feature Interaction Encoder combines this embedding with x. We therefore use learned maps f_{xk} and f_{tk} to transform the covariates and treatment embedding into vectors of the same dimension p

$$r_{xk} = f_{xk}(x). \tag{7}$$

$$r_{tk} = f_{tk}(e_k(t)). \tag{8}$$

Their elementwise product lets treatment information modulate the covariate representation. We retain the original inputs alongside this interaction by concatenation

$$z_k(x,t) = \operatorname{concat}\left[x, e_k(t), r_{xk} \odot r_{tk}\right]. \tag{9}$$

Here $\odot$ denotes elementwise multiplication, and $z_k(x,t) \in \mathbb{R}^{2p+d_t}$ is the encoder output. The parameters of $\mathcal{T}_k$, f_{xk} and f_{tk} form the trainable parameters θ_k of the plugin for anchor k.

Learning through frozen prediction. We consider causal effect foundation models whose key strength is the powerful causal priors acquired during pretraining. These priors can improve the anchor model's outcome predictions and thereby reduce its reference error. To avoid the computational cost of fine tuning the foundation model or repeating pretraining, we keep its pretrained parameters fixed. The encoder then learns through the frozen model's predictions, adapting the inputs while retaining these causal priors.

Let $\mathcal{D}_{\mathrm{tr}} = \{(x_i, t_i, y_i)\}_{i \in I_{\mathrm{tr}}}$ be the training pool, where I_{tr} indexes observations and y_i is the observed outcome. Because the reference error is evaluated at the anchor, we prioritize observations with treatment values near it. We therefore define an unnormalized Gaussian weight for every training observation relative to anchor τ_k

$$\widetilde{w}_{ik} = \left[\exp\left(-(t_i - \tau_k)^2 / (2h_{\mathrm{loc}}^2)\right)\right]^{\rho}. \tag{10}$$

Here $h_{\mathrm{loc}} > 0$ sets the neighborhood scale and $\rho > 0$ controls weight concentration. Gaussian weighting follows the principle of locally weighted regression (Cohn et al., 1996). We normalize over the training pool

$$w_{ik} = (\widetilde{w}_{ik}) / \Big(\sum_{j \in I_{\mathrm{tr}}} \widetilde{w}_{jk}\Big). \tag{11}$$

The weight w_{ik} measures the relevance of training observation i to anchor τ_k.

At each optimization step, we prioritize observations whose treatment values are close to τ_k when training anchor model k. We therefore draw support and query collections S_k and Q_k from the training pool with sampling probabilities proportional to w_{ik} within each side. To represent both binary labels, each collection contains equal numbers of sampled entries from the two sides of the anchor.

To place the support and query collections S_k and Q_k in the same representation, the encoder transforms their covariates and treatments. The resulting support representations $z_k(x_j, t_j)$, together with the binary labels $b_k(t_j)$ and observed outcomes y_j, form the support context

$$\mathcal{C}_k(S_k) = \big((z_k(x_j, t_j), b_k(t_j), y_j)\big)_{j \in S_k}. \tag{12}$$

The causal effect foundation model F_Θ, with fixed pretrained parameters Θ, uses this context to predict a query outcome

$$\widehat{y}_k(x, t \mid S_k) = F_\Theta\big(\mathcal{C}_k(S_k), z_k(x, t), b_k(t)\big). \tag{13}$$

An anchor model consists of the encoder and the frozen predictor F_Θ. The binary input supplies $b_k(t)$, while the feature input supplies the continuous information in $z_k(x, t)$. Query outcomes supervise the encoder through the factual loss

$$\mathcal{L}_{k,\mathrm{fact}} = \Big(\sum_{i \in Q_k} w_{ik}(\widehat{y}_{ki} - y_i)^2\Big) / \Big(\sum_{i \in Q_k} w_{ik}\Big). \tag{14}$$

Here $\widehat{y}_{ki} = \widehat{y}_k(x_i, t_i \mid S_k)$. The squared-error loss also uses the weights from Equation 11, with the denominator normalizing them over Q_k. This observable objective targets local response accuracy. During training, we update the encoder parameters θ_k by backpropagating through F_Θ, while keeping the pretrained parameters Θ unchanged. Once training is complete, the anchor model constructs its prediction context from the complete training pool.

Anchor admission through local weighting. Candidate anchors are treatment values at selected empirical quantile levels q_k in the training pool. Since sampling uses local weights, a few nearby observations may dominate support and query, while more distant observations are less likely to be selected. When replacement is needed, the same observations may appear repeatedly, limiting the distinct information available in both collections. To reduce this loss of sample diversity, we screen each candidate anchor using the effective sample size (ESS) on both sides before sampling. Let $b_{ik} = b_k(t_i)$ be observation i's binary label relative to τ_k. The index set $I_{k,b} = \{i \in I_{\mathrm{tr}} \mid b_{ik} = b\}$ contains the original observations with label b. Their ESS is

$$\mathrm{ESS}_{k,b} = \Big(\sum_{i \in I_{k,b}} w_{ik}\Big)^2 / \Big(\sum_{i \in I_{k,b}} w_{ik}^2\Big). \tag{15}$$

Here $\mathrm{ESS}_{k,0}$ and $\mathrm{ESS}_{k,1}$ measure effective sample sizes below and at or above τ_k, respectively. We retain an anchor only if $\mathrm{ESS}_{k,0} \geq E_{\min}$ and $\mathrm{ESS}_{k,1} \geq E_{\min}$, where $E_{\min} > 0$ is the required threshold for each side.

Smoothness regularization. Equation 5 relates the local error bound to how rapidly predictions change with treatment. To discourage abrupt changes in the prediction slope, we add a supplementary penalty based on predictions at three neighboring treatment values. With covariates x_i and context S_k fixed, let $t_i^{\pm} = t_i \pm \Delta$ and $\widehat{y}_{ki}^{\pm} = \widehat{y}_k(x_i, t_i^{\pm} \mid S_k)$, where $\Delta > 0$ is the treatment increment. Each neighboring treatment uses its corresponding binary label. The penalty is

$$\mathcal{L}_s = \frac{1}{|Q_k|} \sum_{i \in Q_k} \left(\widehat{y}_{ki}^{-} - 2\widehat{y}_{ki} + \widehat{y}_{ki}^{+} \right)^2 . \tag{16}$$

Here $|Q_k|$ counts query entries, including any repeated observations. The second difference compares the prediction change from t_i^- to t_i with the change from t_i to t_i^+. The penalty discourages differences between these two prediction increments, $\widehat{y}_{ki} - \widehat{y}_{ki}^{-}$ and $\widehat{y}_{ki}^{+} - \widehat{y}_{ki}$, encouraging similar slopes over the adjacent treatment intervals. We train the encoder with

$$\mathcal{L}_k = \mathcal{L}_{k,\text{fact}} + \lambda_s \mathcal{L}_s, \tag{17}$$

where $\lambda_s \geq 0$ controls the strength of this supplementary regularization.

3.2 Adaptive Gaussian response aggregation

Using a single anchor model makes response estimation depend entirely on one threshold and that model's local prediction errors. To reduce this dependence across treatment values, we train multiple anchor models with treatment thresholds in $[0, 1]$. For a sample with covariates x and treatment t, relying on one model can make the prediction sensitive to that model's local errors. To reduce this sensitivity, we assign weights to the anchor models and aggregate their predictions into the final response estimate.

Let K be the number of retained anchors, and let $\widehat{y}_k(x, t)$ be the prediction from anchor model k using the complete training context. For a requested treatment t, assign each model a nonnegative, unnormalized aggregation weight $G_k(t)$, with $\sum_{j=1}^{K} G_j(t) > 0$. We normalize these weights to obtain

$$\pi_k(t) = (G_k(t)) / \big(\sum_{j=1}^{K} G_j(t) \big). \tag{18}$$

Here $\pi_k(t)$ is the contribution of model k to the combined prediction. The index j runs over all retained models. The weights satisfy $\pi_k(t) \geq 0$ and $\sum_{k=1}^{K} \pi_k(t) = 1$, so the combined prediction is a weighted average

$$\widehat{y}(x, t) = \sum_{k=1}^{K} \pi_k(t) \widehat{y}_k(x, t). \tag{19}$$

The following results relate the error of this weighted estimate to the errors of the individual anchor models.

Theorem 2 (Aggregation error)**.** *For finite scalar predictions and the normalized weights in Equation 18, the error relative to the true conditional response $\mu^\star(x, t)$ satisfies*

$$\big(\mu^\star(x, t) - \widehat{y}(x, t) \big)^2 \leq \sum_{k=1}^{K} \pi_k(t) \big(\mu^\star(x, t) - \widehat{y}_k(x, t) \big)^2 . \tag{20}$$

The corresponding error decomposition and proof are given in Appendix B.2. The right-hand side of Equation 20 is the weighted average of the anchor models' squared prediction errors at the query (x, t). This connects the accuracy of individual anchor models to the accuracy of their combined prediction.

Theorem 3 (Final prediction error)**.** *If each retained anchor model satisfies Theorem 1, then the aggregation in Theorem 2 obeys*

$$\big(\mu^\star(x, t) - \widehat{y}(x, t) \big)^2 \leq 2 \sum_{k=1}^{K} \pi_k(t) \Big[\varepsilon_k(x; t)^2 + \big(L_\mu(x) + L_{k, b_k(t)}(x) \big)^2 |t - \tau_k|^2 \Big]. \tag{21}$$

Here $\varepsilon_k(x; t)$ is the reference error for the selected binary label, defined in Equation 4. The distance $|t - \tau_k|$ separates the requested treatment from its reference anchor.

The proof is given in Appendix B.3.

The first term in Equation 21 is the weighted average of the anchor models' squared reference errors at the query (x, t). The second accounts for how far each model is used from its anchor and how rapidly the responses can vary. Reducing either contribution while holding the other fixed tightens the error bound. Aggregation therefore complements local learning by controlling how strongly each model contributes to these terms.

Gaussian aggregation. Theorem 3 shows that the final prediction error bound depends on $|t - \tau_k|$, the distance from a sample's requested treatment t to anchor τ_k. Specifically, the expression inside the brackets contains $\left(L_\mu(x) + L_{k,b_k(t)}(x)\right)^2 |t - \tau_k|^2$. Smaller treatment distances reduce the squared distance factor in the final prediction error bound. We therefore use treatment distance as a criterion for assigning greater weight to predictions from nearby anchor models. Specifically, a Gaussian kernel assigns aggregation weights according to treatment distance. For a positive bandwidth $h(t)$, the unnormalized Gaussian weight is

$$G_k(t) = \exp\left(-(t - \tau_k)^2 / (2h(t)^2)\right). \tag{22}$$

The bandwidth sets the treatment range over which models receive substantial weight. At a given t, a closer anchor receives a larger Gaussian weight. Normalizing these weights with Equation 18 and combining the predictions with Equation 19 gives the final response estimate. Each contributing model predicts at the same requested treatment t, with its own binary label $b_k(t)$. The weights $\pi_k(t)$ combine trained models during inference, whereas w_{ik} weights training observations within an anchor model.

Adapting the bandwidth. The retained anchors need not be evenly spaced. A fixed treatment range can therefore include many nearby models in one region and few in another. To adapt the aggregation range to this spacing, we order the anchors by distance from t. Let $\tau_{(r)}(t)$ be the anchor with distance rank r, and define

$$d_{(r)}(t) = |t - \tau_{(r)}(t)|. \tag{23}$$

Thus, $d_{(r)}(t)$ is the distance to the rth nearest anchor, and $d_{(1)}(t) \leq \cdots \leq d_{(K)}(t)$. We use the neighborhood rank k_{agg} and a positive scale s_{agg} to set the bandwidth

$$h(t) = s_{\text{agg}} d_{(k_{\text{agg}})}(t). \tag{24}$$

The rank k_{agg} selects which neighboring anchor defines the local spacing. The scale s_{agg} converts that distance into the width of the Gaussian kernel. Dense anchors produce a narrower kernel, while sparse anchors produce a broader kernel that gives more weight to distant models. This supplements Gaussian aggregation with a treatment dependent range over which local predictions are combined.

4 Experiments

We evaluate whether our input plugin can extend frozen causal foundation models to continuous treatment estimation. The experiments compare our method with established estimators on synthetic and semisynthetic benchmarks.

4.1 Experimental setup

Dataset. We follow the continuous treatment benchmarks used by VCNet and ORIC (Nie et al., 2021; Yan et al., 2025). Synthetic contains 700 observations with six covariates. We use 500 observations for training and 200 for testing in each of four settings with $\beta \in \{0.25, 0.5, 0.75, 1\}$. IHDP contains 747 observations with 25 covariates (Hill, 2011). We use 471 observations for training and 276 for testing. News contains 2,993 document representations with 498 covariates (Newman, 2008). We use 2,000 observations for training and 993 for testing. IHDP and News retain observed covariates and simulate treatments and outcomes, which provides known response curves for evaluation. Appendix C describes the data generation processes, splits, and repetition protocols.

Table 1: Comparison on the Synthetic benchmark. Each entry reports the mean and standard deviation of $\sqrt{\text{AMSE}}$ over 100 repetitions.

	Synthetic $\sqrt{\text{AMSE}}$ over 100 repetitions			
Method	$\beta = 0.25$	$\beta = 0.5$	$\beta = 0.75$	$\beta = 1$
KNN	.2271 ± .0263	.2218 ± .0289	.2186 ± .0245	.2324 ± .0247
BART	.2264 ± .0263	.2201 ± .0294	.2238 ± .0275	.2303 ± .0317
GPS	.1985 ± .0286	.1941 ± .0292	.1964 ± .0244	.2001 ± .0284
MLP	.2018 ± .0309	.1998 ± .0304	.2003 ± .0311	.2101 ± .0301
MLP+GPS	.1897 ± .0288	.1854 ± .0247	.1858 ± .0240	.1923 ± .0224
DRNet	.1953 ± .0243	.2002 ± .0225	.1945 ± .0212	.2041 ± .0230
ADMIT	.1573 ± .0267	.1748 ± .0302	.1834 ± .0376	.1686 ± .0311
ACFR	.1395 ± .0218	.1613 ± .0331	.1619 ± .0317	.1584 ± .0242
VCNet	.1241 ± .0289	.1514 ± .0392	.1566 ± .0485	.1384 ± .0367
VCNet+TR	.1130 ± .0363	.1293 ± .0277	.1422 ± .0557	.1197 ± .0327
ORIC	.1101 ± .0289	.1245 ± .0357	.1315 ± .0401	.1179 ± .0297
TAP-CEFM(Do-PFN)	.1076 ± .0302	.1220 ± .0396	.1286 ± .0411	.1145 ± .0358
TAP-CEFM(CausalPFN)	**.1018 ± .0292**	**.1123 ± .0343**	**.1180 ± .0324**	**.1076 ± .0312**

Compared methods. We compare with KNN (Peterson, 2009), BART (Chipman et al., 2010), GPS (Hirano & Imbens, 2004), MLP, and MLP+GPS following the benchmark suites of VCNet and ORIC (Nie et al., 2021; Yan et al., 2025). MLP+GPS follows the GPS+MLP baseline in ORIC, which combines generalized propensity score adjustment with an MLP outcome model. The continuous treatment neural baselines include DRNet (Schwab et al., 2020), ADMIT (Wang et al., 2022), ACFR (Kazemi & Ester, 2024), VCNet and VCNet+TR (Nie et al., 2021), and ORIC (Yan et al., 2025). VCNet is the original treatment varying network, while VCNet+TR adds functional targeted regularization to that model. Appendix C.4 describes the compared methods.

Evaluation Metrics. The average dose response function is the expected potential outcome at a given treatment value. AMSE averages the squared difference between the estimated and true response functions over all evaluation treatments. We report $\sqrt{\text{AMSE}}$, for which lower values indicate better performance. We report means and standard deviations over 100 repetitions for each of the four Synthetic settings and for IHDP, and over 50 repetitions for our method on News. Appendix D provides the metric definition and evaluation grids.

Table 2: Comparison on the semisynthetic benchmarks. Each entry reports the mean and standard deviation of $\sqrt{\text{AMSE}}$ over 100 IHDP repetitions or 50 News splits.

Method	IHDP	News
KNN	.7957 ± .0567	.5842 ± .2040
BART	.6933 ± .0879	.5671 ± .3545
GPS	.6711 ± .0674	.4275 ± .2328
MLP	.6882 ± .0956	.4332 ± .3105
MLP+GPS	.6163 ± .0837	.4065 ± .2056
DRNet	.5739 ± .0237	.2382 ± .0106
ADMIT	.5176 ± .0360	.1875 ± .0330
ACFR	.5097 ± .0498	.1702 ± .0734
VCNet	.4594 ± .0479	.1879 ± .0959
VCNet+TR	.3672 ± .0557	.1646 ± .0613
ORIC	.3591 ± .0334	.1521 ± .0412
TAP-CEFM(Do-PFN)	.3368 ± .0341	.1617 ± .0447
TAP-CEFM(CausalPFN)	**.2852 ± .0292**	**.1210 ± .0409**

4.2 Results and Discussions

Table 1 shows the results of our methods and the compared methods on the synthetic data over 100 repetitions, and Table 2 shows the results on semi-synthetic benchmarks. Among conventional

baselines, ORIC performs best results, followed by VCNet+TR. ORIC aligns covariate distributions across doses, whereas VCNet+TR combines treatment-dependent coefficients with targeted regularization, therefore achieving promising results among the compared methods.

By translating continuous-dose observations into local binary tasks and aggregating anchor predictions, our plugin can reuse pretrained causal knowledge without refitting the foundation model. We observe that TAP-CEFM with CausalPFN achieves the lowest mean error in every setting, and TAP-CEFM with DoPFN also obtains competitive performance. In particular, TAP-CEFM with CausalPFN has lower mean $\sqrt{\text{AMSE}}$ than ORIC in all four Synthetic settings and on IHDP. These observations suggest that our proposed method is able to leverage knowledge learned by the causal effect foundation model with binary treatments to the problem with continuous treatments. In addition, the TAP-CEFM with CausalPFN has lower mean error than the DoPFN-based variant in all six settings. This suggests that prediction accuracy can vary with the causal effect foundation model chosen as the predictor, and a stronger causal effect foundation model is helpful for achieving better performance.

The qualitative ADRF results are shown in Figure 2, including the ground-truth curve and the results of the three methods with the best performance in Tables 1 and 2. The promising performance across synthetic, IHDP, and News are consistent with the benefit of this adaptation under different response functions.

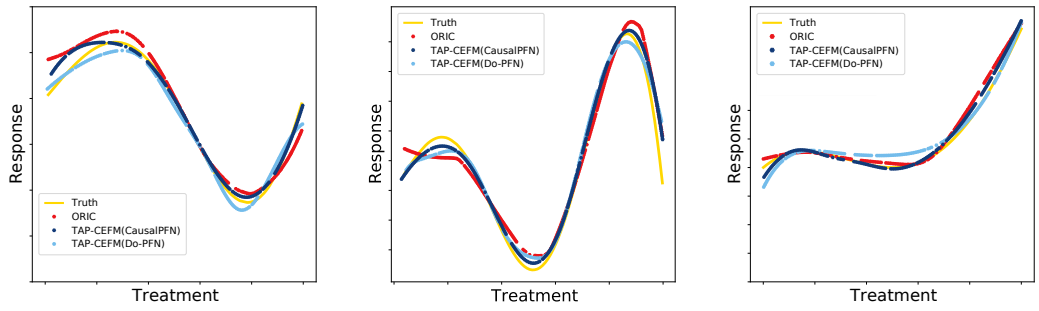


Figure 2: ADRF comparisons on Synthetic, IHDP, and News, from left to right. Yellow lines indicate ground truth; red, dark blue, and light blue markers denote predictions from ORIC, TAP-CEFM (CausalPFN), and TAP-CEFM (DoPFN), respectively.

The parameter sensitivities of the key hyperparameters are empirically evaluated in Appendix F. We further conduct ablation studies in Appendix E by comparing the performance of our method and several variants. We observe that adaptive Gaussian aggregation combines multiple anchor models according to the queried dose and local anchor spacing, thereby achieving better performance compared with that of the single model. We also empirically show the effects of the other components of our methods designed in Section 3. Please refer to the appendices for the detailed descriptions and discussions of the additional experiments.

5 Conclusion

We presented TAP-CEFM, a treatment-aware plugin that adapts frozen causal effect foundation models with binary treatment interfaces to continuous treatment effect estimation. The plugin constructs local binary tasks around treatment anchors, preserves continuous dose information through a trainable encoder, and combines anchor predictions with adaptive Gaussian aggregation. We adopt two causal effect foundation models with binary treatments as the predictor, and conduct experiments on benchmark datasets to evaluate the performance of our method. The results demonstrate the effectiveness of our method and the effects of the components in our method.

AI USE STATEMENT

Generative AI tools were used to improve the language and presentation of the manuscript and to assist with code editing. They were not used to develop research ideas, formulate scientific claims, design the study, analyze data, interpret experimental results, or draw scientific conclusions. All AI-assisted modifications were carefully reviewed by all authors, who take full responsibility for the final content of this work.

REFERENCES

Vahid Balazadeh Meresht, Hamidreza Kamkari, Valentin Thomas, Junwei Ma, Bingru Li, Jesse Cresswell, and Rahul Krishnan. Causalpfn: Amortized causal effect estimation via in-context learning. *Advances in Neural Information Processing Systems*, 38:154945–154984, 2026.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

Hugh A Chipman, Edward I George, and Robert E McCulloch. Bart: Bayesian additive regression trees. 2010.

David A Cohn, Zoubin Ghahramani, and Michael I Jordan. Active learning with statistical models. *Journal of artificial intelligence research*, 4:129–145, 1996.

Benjamin Feuer, Robin T Schirrmeister, Valeriia Cherepanova, Chinmay Hegde, Frank Hutter, Micah Goldblum, Niv Cohen, and Colin White. Tunetables: Context optimization for scalable prior-data fitted networks. *Advances in Neural Information Processing Systems*, 37:83430–83464, 2024.

Christian Fong, Chad Hazlett, and Kosuke Imai. Covariate balancing propensity score for a continuous treatment: Application to the efficacy of political advertisements. *The Annals of Applied Statistics*, 12(1):156–177, 2018.

Jennifer L Hill. Bayesian nonparametric modeling for causal inference. *Journal of Computational and Graphical Statistics*, 20(1):217–240, 2011.

Keisuke Hirano and Guido W. Imbens. *The Propensity Score with Continuous Treatments*, chapter 7, pp. 73–84. John Wiley & Sons, Ltd, 2004. ISBN 9780470090459. doi: 10.1002/0470090456.ch7.

Noah Hollmann, Samuel Müller, Katharina Eggensperger, and Frank Hutter. Tabpfn: A transformer that solves small tabular classification problems in a second. *arXiv preprint arXiv:2207.01848*, 2022.

Noah Hollmann, Samuel Müller, Lennart Purucker, Arjun Krishnakumar, Max Körfer, Shi Bin Hoo, Robin Tibor Schirrmeister, and Frank Hutter. Accurate predictions on small data with a tabular foundation model. *Nature*, 637(8045):319–326, 2025.

Kosuke Imai and David A Van Dyk. Causal inference with general treatment regimes: Generalizing the propensity score. *Journal of the American Statistical Association*, 99(467):854–866, 2004.

Guido W Imbens and Donald B Rubin. Causal inference in statistics, social, and biomedical sciences. *New York*, 517, 2015.

Amirreza Kazemi and Martin Ester. Adversarially balanced representation for continuous treatment effect estimation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 13085–13093, 2024.

Edward H Kennedy, Zongming Ma, Matthew D McHugh, and Dylan S Small. Non-parametric methods for doubly robust estimation of continuous treatment effects. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 79(4):1229–1245, 2017.

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.

Si-Yang Liu and Han-Jia Ye. Tabpfn unleashed: A scalable and effective solution to tabular classification problems. *arXiv preprint arXiv:2502.02527*, 2025.

Samuel Müller, Noah Hollmann, Sebastian Pineda Arango, Josif Grabocka, and Frank Hutter. Transformers can do bayesian inference. *arXiv preprint arXiv:2112.10510*, 2021.

David Newman. Bag of Words. UCI Machine Learning Repository, 2008. DOI: https://doi.org/10.24432/C5ZG6P.

Lizhen Nie, Mao Ye, Qiang Liu, and Dan Nicolae. Vcnet and functional targeted regularization for learning causal effects of continuous treatments. *arXiv preprint arXiv:2103.07861*, 2021.

Judea Pearl. Causal inference in statistics: An overview. 2009.

Leif E Peterson. K-nearest neighbor. *Scholarpedia*, 4(2):1883, 2009.

Jingang Qu, David Holzmüller, Gaël Varoquaux, and Marine Le Morvan. Tabicl: A tabular foundation model for in-context learning on large data. *arXiv preprint arXiv:2502.05564*, 2025.

Jake Robertson, Arik Reuter, Siyuan Guo, Noah Hollmann, Frank Hutter, and Bernhard Schölkopf. Do-pfn: In-context learning for causal effect estimation. *Advances in Neural Information Processing Systems*, 38:174811–174848, 2026.

Paul R Rosenbaum and Donald B Rubin. The central role of the propensity score in observational studies for causal effects. *Biometrika*, 70(1):41–55, 1983.

Donald B Rubin. Estimating causal effects of treatments in randomized and nonrandomized studies. *Journal of educational Psychology*, 66(5):688, 1974.

Patrick Schwab, Lorenz Linhardt, Stefan Bauer, Joachim M Buhmann, and Walter Karlen. Learning counterfactual representations for estimating individual dose-response curves. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pp. 5612–5619, 2020.

Uri Shalit, Fredrik D Johansson, and David Sontag. Estimating individual treatment effect: generalization bounds and algorithms. In *International conference on machine learning*, pp. 3076–3085. PMLR, 2017.

Claudia Shi, David Blei, and Victor Veitch. Adapting neural networks for the estimation of treatment effects. *Advances in neural information processing systems*, 32, 2019.

Valentin Thomas, Junwei Ma, Rasa Hosseinzadeh, Keyvan Golestan, Guangwei Yu, Maksims Volkovs, and Anthony Caterini. Retrieval & fine-tuning for in-context tabular models. *Advances in Neural Information Processing Systems*, 37:108439–108467, 2024.

Stefan Wager and Susan Athey. Estimation and inference of heterogeneous treatment effects using random forests. *Journal of the American Statistical Association*, 113(523):1228–1242, 2018.

Xin Wang, Shengfei Lyu, Xingyu Wu, Tianhao Wu, and Huanhuan Chen. Generalization bounds for estimating causal effects of continuous treatments. *Advances in Neural Information Processing Systems*, 35:8605–8617, 2022.

Zifeng Wang and Jimeng Sun. Transtab: Learning transferable tabular transformers across tables. *Advances in Neural Information Processing Systems*, 35:2902–2915, 2022.

Yuguang Yan, Zongyu Li, Haolin Yang, Zeqin Yang, Hao Zhou, Ruichu Cai, and Zhifeng Hao. Reducing confounding bias without data splitting for causal inference via optimal transport. In *Forty-second International Conference on Machine Learning*, 2025.

Bingzhao Zhu, Xingjian Shi, Nick Erickson, Mu Li, George Karypis, and Mahsa Shoaran. Xtab: Cross-table pretraining for tabular transformers. *arXiv preprint arXiv:2305.06090*, 2023.

A PRELIMINARIES

We first state the continuous treatment causal target and its three identification assumptions, then describe the causal effect foundation models used in our method.

A.1 CONTINUOUS TREATMENT CAUSAL ESTIMATION

Let $X \in \mathbb{R}^p$ be pretreatment covariates, $T \in [0, 1]$ a dose, and $Y(t)$ the potential outcome under dose t. Observational data are

$$\mathcal{D}_{\text{obs}} = \{(x_i, t_i, y_i)\}_{i=1}^n, \qquad y_i = Y_i(t_i). \tag{25}$$

Only one potential outcome is observed per unit. Unlike supervised prediction of factual labels, causal estimation asks what would happen under doses that the unit did not receive. Our targets are the conditional response and the average dose response function (ADRF)

$$\mu^\star(x, t) = \mathbb{E}[Y(t) \mid X = x], \qquad \mu(t) = \mathbb{E}_X[\mu^\star(X, t)]. \tag{26}$$

The causal interpretation requires three standard assumptions.

Assumption A.1 (Consistency). The observed outcome equals the potential outcome under the received dose. Thus, $Y = Y(T)$.

Assumption A.2 (Conditional ignorability). The potential outcome at every dose is independent of the observed treatment after conditioning on pretreatment covariates. Thus, $Y(t) \perp\!\!\!\perp T \mid X$ for every $t \in [0, 1]$.

Assumption A.3 (Positivity). Let $p(t \mid x)$ denote the conditional treatment density. This density satisfies $p(t \mid x) > 0$ at every covariate and dose pair in the target support.

Under these assumptions, the conditional response is identified by

$$\mu^\star(x, t) = \mathbb{E}[Y \mid X = x, T = t]. \tag{27}$$

The three assumptions allow outcome regression to estimate a causal response. However, factual accuracy alone does not guarantee accurate counterfactual predictions because observational treatment assignment can leave some doses poorly supported (Rubin, 1974; Hirano & Imbens, 2004; Kennedy et al., 2017).

A.2 CAUSAL EFFECT FOUNDATION MODELS

Causal effect foundation models such as CausalPFN and DoPFN learn causal priors during pretraining on simulated tasks (Balazadeh Meresht et al., 2026; Robertson et al., 2026). Through in context learning, they use observational examples to predict potential outcomes without updating their pretrained parameters. We represent their inputs as a labeled observational context $\mathcal{C} = \{(u_i, a_i, y_i)\}_{i=1}^{N_s}$ and a query (u_q, a_q). Here u denotes a covariate representation, a a binary treatment label, y_i a factual outcome, and N_s the context size. With fixed pretrained parameters Θ, we write the point prediction as

$$F_\Theta(\mathcal{C}; u_q, a_q), \tag{28}$$

which estimates the expected potential outcome for the query. This interface gives covariates, treatment labels, and factual outcomes distinct roles. A general tabular PFN can serve as an outcome regressor under the assumptions in Appendix A.2, but its factual prediction pretraining does not explicitly target counterfactual responses (Hollmann et al., 2022). Causal effect foundation models instead acquire priors tailored to causal estimation. Our plugin targets the binary treatment interfaces of the frozen predictors, which require adaptation before they can accept continuous treatment data.

B PROOFS OF THE THEORETICAL RESULTS

We prove the three theorems in Section 3 in their order of appearance, using the notation and assumptions stated in the main text.

B.1 LOCAL PREDICTION ERROR

Proof of Theorem 1. For a query (x, t) and anchor k, let $b = b_k(t)$. Then t and τ_k both belong to $J_{k,b}$, and $\widehat{y}_k(x,t) = \widehat{y}_{k,b}(x,t)$. The reference prediction at τ_k uses this same binary input b, as specified in Equation 4. In particular, it is $\widehat{y}_{k,0}(x,\tau_k)$ when $t < \tau_k$, and $\widehat{y}_{k,1}(x,\tau_k)$ otherwise.

Adding and subtracting the true response and the selected prediction at the anchor, the triangle inequality gives

$$\begin{aligned}|\mu^\star(x,t) - \widehat{y}_k(x,t)| \le\ & |\mu^\star(x,t) - \mu^\star(x,\tau_k)| \\ & + |\mu^\star(x,\tau_k) - \widehat{y}_{k,b}(x,\tau_k)| \\ & + |\widehat{y}_{k,b}(x,\tau_k) - \widehat{y}_{k,b}(x,t)|.\end{aligned}$$

The middle term equals $\varepsilon_k(x;t)$. By Equations 2 and 3, the first and third terms are bounded by $L_\mu(x)|t-\tau_k|$ and $L_{k,b}(x)|t-\tau_k|$, respectively. Consequently,

$$\begin{aligned}|\mu^\star(x,t) - \widehat{y}_k(x,t)| &\le L_\mu(x)|t-\tau_k| + \varepsilon_k(x;t) + L_{k,b}(x)|t-\tau_k| \\ &= \varepsilon_k(x;t) + \big(L_\mu(x) + L_{k,b_k(t)}(x)\big)|t-\tau_k|.\end{aligned}$$

This is the bound in Equation 5. □

B.2 AGGREGATION ERROR

Proof of Theorem 2. At a query (x,t), the normalized weights satisfy $\pi_k(t) \ge 0$ and $\sum_{k=1}^K \pi_k(t) = 1$, while $\widehat{y}(x,t) = \sum_{k=1}^K \pi_k(t)\widehat{y}_k(x,t)$. It follows that the weighted deviations from the aggregate sum to zero:

$$\sum_{k=1}^K \pi_k(t)\big(\widehat{y}_k(x,t) - \widehat{y}(x,t)\big) = \sum_{k=1}^K \pi_k(t)\widehat{y}_k(x,t) - \widehat{y}(x,t)\sum_{k=1}^K \pi_k(t) = 0.$$

Using

$$\mu^\star(x,t) - \widehat{y}_k(x,t) = \big(\mu^\star(x,t) - \widehat{y}(x,t)\big) - \big(\widehat{y}_k(x,t) - \widehat{y}(x,t)\big)$$

and expanding the square therefore yields

$$\begin{aligned}&\sum_{k=1}^K \pi_k(t)\big(\mu^\star(x,t) - \widehat{y}_k(x,t)\big)^2 \\ &= \big(\mu^\star(x,t) - \widehat{y}(x,t)\big)^2 \sum_{k=1}^K \pi_k(t) \\ &\quad - 2\big(\mu^\star(x,t) - \widehat{y}(x,t)\big)\sum_{k=1}^K \pi_k(t)\big(\widehat{y}_k(x,t) - \widehat{y}(x,t)\big) \\ &\quad + \sum_{k=1}^K \pi_k(t)\big(\widehat{y}_k(x,t) - \widehat{y}(x,t)\big)^2 \\ &= \big(\mu^\star(x,t) - \widehat{y}(x,t)\big)^2 + \sum_{k=1}^K \pi_k(t)\big(\widehat{y}_k(x,t) - \widehat{y}(x,t)\big)^2.\end{aligned}$$

Rearranging gives the exact error decomposition

$$\begin{aligned}\big(\mu^\star(x,t) - \widehat{y}(x,t)\big)^2 = &\sum_{k=1}^K \pi_k(t)\big(\mu^\star(x,t) - \widehat{y}_k(x,t)\big)^2 \\ &- \sum_{k=1}^K \pi_k(t)\big(\widehat{y}_k(x,t) - \widehat{y}(x,t)\big)^2.\end{aligned} \tag{29}$$

Since every term in the second sum is nonnegative, the identity implies

$$\big(\mu^\star(x,t) - \widehat{y}(x,t)\big)^2 \le \sum_{k=1}^K \pi_k(t)\big(\mu^\star(x,t) - \widehat{y}_k(x,t)\big)^2,$$

which is Equation 20. □

B.3 Final prediction error

Proof of Theorem 3. Theorem 2 bounds the aggregate squared error by the weighted average of the individual squared errors:

$$\big(\mu^\star(x,t)-\widehat{y}(x,t)\big)^2 \le \sum_{k=1}^{K}\pi_k(t)\big(\mu^\star(x,t)-\widehat{y}_k(x,t)\big)^2.$$

For each retained anchor, define the nonnegative distance contribution

$$D_k(x,t)=\big(L_\mu(x)+L_{k,b_k(t)}(x)\big)|t-\tau_k|.$$

Theorem 1 gives $|\mu^\star(x,t)-\widehat{y}_k(x,t)|\le\varepsilon_k(x;t)+D_k(x,t)$. Both terms on the right are nonnegative, so squaring preserves the inequality. Applying $(a+b)^2\le 2a^2+2b^2$, which follows from $(a-b)^2\ge 0$, then gives

$$\begin{aligned}\big(\mu^\star(x,t)-\widehat{y}_k(x,t)\big)^2 &\le \big(\varepsilon_k(x;t)+D_k(x,t)\big)^2\\ &\le 2\varepsilon_k(x;t)^2+2D_k(x,t)^2.\end{aligned}$$

Multiplying by the nonnegative weights $\pi_k(t)$, summing over the retained anchors, and substituting into the aggregate bound yields

$$\begin{aligned}&\big(\mu^\star(x,t)-\widehat{y}(x,t)\big)^2\\ &\le 2\sum_{k=1}^{K}\pi_k(t)\big[\varepsilon_k(x;t)^2+D_k(x,t)^2\big]\\ &= 2\sum_{k=1}^{K}\pi_k(t)\Big[\varepsilon_k(x;t)^2+\big(L_\mu(x)+L_{k,b_k(t)}(x)\big)^2|t-\tau_k|^2\Big].\end{aligned}$$

This proves Equation 21. □

C Benchmark and Evaluation Details

This appendix records the data generation processes and evaluation measure used in Section 4. In every Gaussian distribution below, the second argument denotes the variance. Let $\sigma(a)=(1+\exp(-a))^{-1}$.

C.1 Synthetic data generation process

We use the continuous treatment benchmark of VCNet and ORIC (Nie et al., 2021; Yan et al., 2025). Each repetition contains 500 training and 200 test observations. We regenerate the covariates, treatments, and outcomes for each seed and evaluate 100 repetitions for each value of β. This parameter enters both the treatment assignment and the conditional response functions below. The treatment and outcome noises are independent.

For each observation, the six covariates are sampled independently as $x_{ij}\sim\mathrm{Unif}(0,1)$. For $\beta\in\{0.25,0.5,0.75,1\}$, define

$$\begin{aligned}s_\beta(x_i)=&\frac{10\sin(\max\{x_{i1},x_{i2},x_{i3}\})+\max\{x_{i3},x_{i4},x_{i5}\}^3}{1+(x_{i1}+x_{i5})^2}\\ &+\sin(\beta x_{i3})\left[1+\exp(x_{i4}-\beta x_{i3})\right]+x_{i3}^2+2\sin(x_{i4})+2x_{i5}-6.5, \qquad (30)\\ t_i=&\,\sigma(s_\beta(x_i)+\epsilon_{ti}), \qquad \epsilon_{ti}\sim\mathcal{N}(0,0.25). \qquad (31)\end{aligned}$$

The conditional response and observed outcome are

$$\begin{aligned}\mu_{\mathrm{S}}(x_i,t)=&\,\cos(2\pi(t-\beta))\left[t^2+\frac{4\max\{x_{i1},x_{i6}\}^3\sin(x_{i4})}{1+2x_{i3}^2}\right], \qquad (32)\\ y_i=&\,\mu_{\mathrm{S}}(x_i,t_i)+\epsilon_{yi}, \qquad \epsilon_{yi}\sim\mathcal{N}(0,0.25). \qquad (33)\end{aligned}$$

C.2 IHDP DATA GENERATION PROCESS

IHDP contains 747 observations with 25 covariates measured before treatment (Hill, 2011). We retain the observed covariates and simulate continuous treatments and outcomes using the benchmark generator below. Each repetition contains 471 training and 276 test observations. We regenerate treatments, outcomes, and the split for each repetition, while keeping the covariates fixed. The data generation and split seeds range from 42 to 141, which gives 100 repetitions. The treatment score and outcome receive independent Gaussian noise.

Each IHDP covariate column is transformed as

$$x_{ij} \leftarrow \frac{x_{ij} - \min_\ell x_{\ell j}}{\max_\ell x_{\ell j}}, \tag{34}$$

with a denominator of one for a numerically degenerate column. Define

$$\mathcal{S}_1 = \{4, 7, \ldots, 15\}, \qquad \mathcal{S}_2 = \{16, 17, \ldots, 25\}, \tag{35}$$

and

$$m_{ig} = \frac{1}{|\mathcal{S}_g|} \sum_{j \in \mathcal{S}_g} x_{ij}, \qquad c_1 = \frac{1}{n} \sum_{i=1}^{n} m_{i1}, \qquad c_2 = c_1. \tag{36}$$

With $\widetilde{\pi} = 3.14159$, treatment and outcome are generated by

$$a_i = \frac{x_{i1}}{1 + x_{i2}} + \frac{\max\{x_{i3}, x_{i5}, x_{i6}\}}{0.2 + \min\{x_{i3}, x_{i5}, x_{i6}\}} + \tanh(5[m_{i2} - c_2]) - 2, \tag{37}$$

$$t_i = \sigma(2[a_i + \epsilon_{ti}]), \qquad \epsilon_{ti} \sim \mathcal{N}(0, 0.25), \tag{38}$$

$$\mu_{\mathrm{I}}(x_i, t) = \frac{\sin(3\widetilde{\pi} t)}{1.2 - t} \left[1.5 \tanh(5[m_{i1} - c_1]) + \frac{0.5 \exp(0.2[x_{i1} - x_{i6}])}{0.1 + \min\{x_{i2}, x_{i3}, x_{i5}\}}\right], \tag{39}$$

$$y_i = \mu_{\mathrm{I}}(x_i, t_i) + \epsilon_{yi}, \qquad \epsilon_{yi} \sim \mathcal{N}(0, 0.25). \tag{40}$$

C.3 NEWS DATA GENERATION PROCESS

News contains 2,993 document representations from the New York Times corpus (Newman, 2008). We generate one realization of continuous treatments and outcomes and keep it fixed across repetitions. Each split contains 2,000 training and 993 test observations. Repetitions vary the split and optimization seed. Our method uses 50 splits, while the rerun baselines use 20. This difference in repetition budgets also applies to the summaries in the main text.

Every nonzero News feature column is divided by its column maximum. Using a fixed seed, we draw $v_1, v_2, v_3 \sim \mathcal{N}(0, I)$ and normalize each vector to unit Euclidean norm. Define the stabilized denominator

$$d(z) = \begin{cases} 10^{-8}, & |z| < 10^{-8}, \\ z, & \text{otherwise.} \end{cases} \tag{41}$$

The treatment and outcome are generated as

$$\alpha_i = \left|\frac{x_i^\top v_3}{d(2 x_i^\top v_2)}\right| + 10^{-4}, \qquad t_i \sim \mathrm{Beta}(2, \alpha_i), \tag{42}$$

$$u_i = \mathrm{clip}_{[-2,2]}\left\{\exp\left[\min\left(\frac{x_i^\top v_2}{d(x_i^\top v_3)} - 0.3, \log 2\right)\right]\right\}, \tag{43}$$

$$\mu_{\mathrm{N}}(x_i, t) = 2\left[u_i + 20 x_i^\top v_1\right]\left[4(t - 0.5)^2 \sin\left(\frac{\pi t}{2}\right)\right], \tag{44}$$

$$y_i = \mu_{\mathrm{N}}(x_i, t_i) + \epsilon_i, \qquad \epsilon_i \sim \mathcal{N}(0, 0.5). \tag{45}$$

C.4 COMPARED METHODS

The statistical baselines are KNN (Peterson, 2009), Bayesian additive regression trees (BART) (Chipman et al., 2010), and generalized propensity scores (GPS) (Hirano & Imbens, 2004). KNN

uses nearby observations for prediction, while BART combines regression trees to model nonlinear outcomes. GPS adjusts for covariate dependent treatment assignment through the conditional treatment density. MLP provides a neural outcome regression baseline. MLP+GPS follows the GPS+MLP implementation in ORIC, which uses an MLP to strengthen the outcome model associated with GPS. We retain the MLP+GPS label used in our result tables. We follow the benchmark implementations used by VCNet and ORIC (Nie et al., 2021; Yan et al., 2025).

The remaining baselines are designed for continuous treatment estimation. DRNet uses treatment specific prediction heads (Schwab et al., 2020). ADMIT and ACFR learn representations that address covariate imbalance (Wang et al., 2022; Kazemi & Ester, 2024). VCNet uses coefficients that vary with treatment. VCNet denotes the original model, while VCNet+TR adds functional targeted regularization to the same backbone (Nie et al., 2021). ORIC uses optimal transport to align covariate distributions without partitioning observations into separate groups for distribution estimation (Yan et al., 2025). TAP-CEFM instead trains an input plugin through a frozen causal predictor. CausalPFN is the main predictor, and the experiments also evaluate the plugin with frozen DoPFN (Robertson et al., 2026).

D Evaluation Measure Details

We evaluate predictions at the observed treatment values in each test set. For Synthetic, the true ADRF is obtained by averaging the known conditional response over the test covariates. For IHDP and News, we use the benchmark files that provide the test treatment grid and corresponding true ADRF. Predictions are averaged over the evaluation covariates at every treatment value. Within shared repetitions, methods use the same data realization, training and test split, evaluation locations, and true curve. The different News repetition budgets therefore do not imply that every reported summary contains exactly the same set of repetitions.

At treatment s, the ADRF estimate in repetition r is

$$\widehat{\mu}_r(s) = \frac{1}{n_{\text{te}}} \sum_{i \in I_{\text{te}}} \widehat{y}_r(x_i, s). \tag{46}$$

Let $\mathcal{G}_r = \{s_{r1}, \ldots, s_{rM_r}\}$ denote the evaluation grid. The reported error for each repetition is

$$\text{AMSE}_r = \frac{1}{M_r} \sum_{m=1}^{M_r} \left[\widehat{\mu}_r(s_{rm}) - \mu_r(s_{rm})\right]^2, \tag{47}$$

$$e_r = \sqrt{\text{AMSE}_r}. \tag{48}$$

Across R repetitions, tables report $\overline{e} \pm s_e$, where

$$\overline{e} = \frac{1}{R} \sum_{r=1}^{R} e_r, \qquad s_e = \left[\frac{1}{R-1} \sum_{r=1}^{R} (e_r - \overline{e})^2\right]^{1/2}. \tag{49}$$

E IHDP Component Ablations

Table 3 reports the IHDP Full result from Table 2 alongside seven component ablations. Each variant changes one mechanism of the treatment aware plugin while retaining the frozen CausalPFN predictor and the remaining components. The entries summarize repeated $\sqrt{\text{AMSE}}$ results. Lower values indicate more accurate ADRF estimates.

The **Single anchor** variant retains the admitted anchor nearest the treatment median. **No treatment adapter** replaces the learned adapter with its three raw inputs t, τ_k, and $t - \tau_k$.

No soft weighting removes local Gaussian weights from episode sampling and factual loss while retaining the original ESS anchor admission. **No feature interaction** concatenates the original covariates and treatment embedding without the learned multiplicative interaction.

No ESS admission disables the anchor filter based on independent sample counts and weighted ESS on both sides. **Fixed Gaussian** replaces adaptive aggregation with a fixed Gaussian bandwidth of

Table 3: IHDP component ablation results with a frozen CausalPFN predictor. Values are the mean and standard deviation of $\sqrt{\text{AMSE}}$ across repetitions. Lower values indicate more accurate ADRF estimates.

Variant	IHDP $\sqrt{\text{AMSE}}$
Full	**.2852 ± .0292**
Single anchor	.5062 ± .0389
No treatment adapter	.4407 ± .0281
No soft weighting	.4942 ± .0354
No feature interaction	.4354 ± .0276
No ESS admission	.4176 ± .0349
Fixed Gaussian	.4582 ± .0297
No smoothness loss	.3859 ± .0386

0.10. **No smoothness loss** sets $\lambda_s = 0$ in Equation 17, removing the local smoothness loss while retaining factual supervision.

The single anchor variant retains a continuous treatment representation, so its response curve need not be a step function. Full provides the reference for the seven component variants. The available summaries do not establish matched repetitions for paired component comparisons.

We repeat the first six component tests with a frozen DoPFN predictor. In this case, Full and every variant use the same repetitions, treatment generation, data splits, candidate anchors, and evaluation grid. Each variant changes the named component while retaining the DoPFN specific Full configuration. The results appear in Table 4.We also report No final ADRF smoothness, which removes the final response smoothness penalty from the DoPFN configuration.

Table 4: IHDP component ablations with a frozen DoPFN predictor. Values are the mean and standard deviation of $\sqrt{\text{AMSE}}$ across matched repetitions. Lower values indicate more accurate ADRF estimates.

Variant	IHDP $\sqrt{\text{AMSE}}$
Full	**.3368 ± .0341**
Single anchor	.6557 ± .0398
No treatment adapter	.5230 ± .0376
No soft weighting	.5538 ± .0384
No feature interaction	.6072 ± .0393
No ESS admission	.4638 ± .0417
Fixed Gaussian	.4585 ± .0425
No final ADRF smoothness	.4293 ± .0416

Every completed DoPFN ablation increases the estimation error. The single anchor variant produces the largest degradation because one local comparison cannot represent the complete treatment continuum. Removing feature interaction, soft weighting, or the treatment adapter also causes substantial degradation, which shows that DoPFN depends on the plugin to construct useful local causal tasks. Disabling ESS admission increases the average number of experts from (8.86) to (13.31), while the error rises from (0.3368) to (0.4638). This result indicates that admitting more anchors does not help when some anchors lack effective support on both sides. Fixed Gaussian aggregation also performs worse than adaptive aggregation, which supports a treatment dependent expert combination. Removing the final ADRF smoothness penalty also yields a higher mean error than the DoPFN Full configuration.

F IHDP Hyperparameter Sensitivity

We examine six core hyperparameters on IHDP with a one factor at a time analysis. Each setting changes one value from the reference configuration and keeps the remaining model choices fixed. Every point uses the same 20 seeds from 42 to 61. Figure 3 reports the mean $\sqrt{\text{AMSE}}$, while the error bars show 95% confidence intervals for the mean across repetitions.

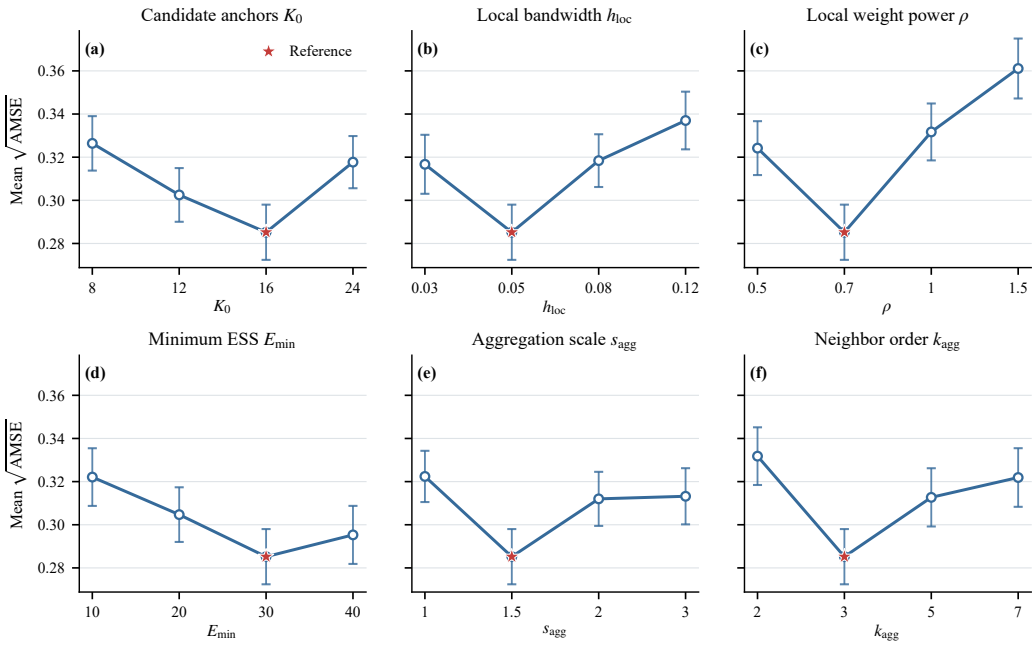


Figure 3: Hyperparameter sensitivity on IHDP. We vary the candidate anchor count K_0, local Gaussian bandwidth h_{loc}, local weight power ρ, minimum two sided effective sample size E_{min}, adaptive aggregation scale s_{agg}, and aggregation neighbor order k_{agg}. Each point reports the mean over 20 matched repetitions. Error bars show 95% confidence intervals for the mean. Red stars mark the reference configuration. Lower values are better.

The reference values achieve the lowest mean error in all six panels. The curves show that too few or too many anchors reduce accuracy, while overly narrow or broad localization weakens the balance between local relevance and effective support. The model is most sensitive to large local weight powers, which concentrate influence on a small number of observations. The similar results for $E_{\text{min}} = 30$ and $E_{\text{min}} = 40$ indicate a comparatively stable region for the expert admission threshold. Aggregation also favors an intermediate neighborhood, since a narrow neighborhood relies on too few experts and a broad neighborhood mixes less relevant experts.