DRIFT-AWARE UNCERTAINTY QUANTIFICATION VIA A FUNCTIONAL SPECTRAL-NEWTON METHOD

Thiago R. Ramos
Federal University of São Carlos
thiagorr@ufscar.br

Alek Fröhlich
CSML, Istituto Italiano di Tecnologia
University of Genoa
alek.frohlich@iit.it

Daniel Perazzo
CSML, Istituto Italiano di Tecnologia
University of Genoa
daniel.rodrigues@iit.it

Massimiliano Pontil
CSML, Istituto Italiano di Tecnologia
University College London
massimiliano.pontil@iit.it

ABSTRACT

Machine learning models are increasingly deployed in high-risk domains such as healthcare and finance, where uncertainty quantification is essential and distribution shifts can severely degrade predictive performance. However, many existing approaches to shift detection and adaptation address isolated components of the *catch–adapt–operate* cycle, often without explicitly accounting for predictive uncertainty. In this paper, we introduce a stagewise framework for learning conditional distributions that directly targets harmful changes affecting predictive performance. Our method learns a spectral decomposition of the density ratio $f_{XY}/(f_X f_Y)$ via alternating functional Newton updates, reminiscent of gradient boosting methods. We also introduce a performance degradation metric for identifying shifts that are harmful and should trigger adaptation. Preliminary experiments on conditional distribution estimation benchmarks with induced shifts suggest that this approach offers a principled path toward robust conditional distribution modeling in high-risk, nonstationary environments.

1 INTRODUCTION

Machine learning systems have achieved remarkable predictive performance and are increasingly deployed in high-risk settings such as healthcare and finance (Fröhlich et al., 2025; Csillag et al., 2023; Chernozhukov et al., 2021). In these domains, decisions informed by model predictions often carry significant consequences, making reliability and calibrated uncertainty quantification essential (Angelopoulos & Bates, 2023). However, once deployed, such systems may fail *silently* due to their reliance on the joint data-generating distribution P_{XY}, which can evolve over time and differ substantially from the development environment (Rabanser et al., 2019; Steyerberg, 2019; Gibbs & Candes, 2021; Quinonero-Candela et al., 2022). As a result, models that perform well at training time may exhibit degraded reliability, miscalibration, or loss of predictive validity after deployment (Finlayson et al., 2021; Van Calster et al., 2023).

In this work, we propose a methodology for learning conditional distributions in an incremental, stagewise manner that explicitly characterizes when distributional changes lead to performance degradation and how models should be updated in response. Our approach builds on functional gradient methods (Grubb & Bagnell, 2012; Luenberger, 1969), enabling controlled updates in function space when degradation is detected. A key property of this formulation is that, at convergence, the functional gradients vanish; thus, when the model is re-evaluated on new data from a shifted distribution, large gradient norms provide a direct, interpretable signal that the learned conditional structure no longer fits the data—without requiring a separate drift detection mechanism. Through empirical studies, we demonstrate that the proposed method can reliably detect harmful drift and perform targeted updates that restore predictive performance while avoiding unnecessary adaptation.

Contributions. Our main contributions are as follows:

- We introduce a new algorithm for learning conditional distributions $Y|X$ via a spectral decomposition of the density ratio $\kappa(x, y) = f_{XY}(x, y)/f_X(x)f_Y(y)$ using alternating functional Newton updates.
- We describe how norms of functional gradients provide a signal for detecting harmful distribution shifts and guiding model adaptation.
- We validate our method on the conditional density estimation benchmark from (Rothfuss et al., 2019) modified by introducing distribution shifts.

Paper organization. Section 2 reviews related work on spectral representation learning, functional optimization, and dataset shift. Section 3 introduces integral operators, their low-rank decompositions, and functional optimization via Gateaux derivatives. Section 4 presents our proposed functional spectral–Newton method (**FSNM**). Section 5 develops a framework for detecting and adapting to harmful distribution drift using norms of functional gradients. Section 6 reports numerical experiments. Section 7 concludes the paper. All proofs are deferred to the appendix.

2 Related Work

Spectral representation learning. Expansions in orthogonal bases such as Fourier, wavelets, and smoothing splines have a long history in nonparametric statistical inference (Efromovich, 1999). While these bases are well understood, they perform poorly in high dimensions and lack adaptivity to the data distribution. This limitation motivated the use of bases derived from the spectral decomposition of linear integral operators defined by symmetric positive semi-definite kernels (Izbicki, 2014). More recently, research has shifted toward learned spectral bases, or features, which have been applied across diverse settings including dynamical systems (Turri et al., 2025), reinforcement learning (Hu et al., 2024), and nonparametric causal inference (Wang et al., 2022; Sun et al., 2025; Meunier et al., 2025b;a; Fröhlich et al., 2025). Many of these methods resemble contrastive approaches in self-supervised learning (HaoChen et al., 2021). In cases where the objective is to learn linear integral operators, the problem often reduces to estimating their associated kernel (Kostic et al., 2024; Ordoñez-Apraez et al., 2025); for the conditional expectation operator, this kernel ($\kappa = f_{XY}/(f_X f_Y)$) can be expressed as a density ratio, for which a variety of estimation strategies have been developed (Sugiyama et al., 2012).

Functional optimization. Optimization in infinite-dimensional vector spaces has a classical treatment in the mathematical programming literature (Luenberger, 1969). In machine learning, this viewpoint entered prominently through boosting, which can be interpreted as functional gradient descent in a space of functions (Mason et al., 1999; Friedman, 2001; Schapire & Freund, 2012; Bühlmann & Hothorn, 2007). Related perspectives cast boosting as an iterative procedure for minimizing convex objectives in function space (Grubb & Bagnell, 2012). In our setting, adopting this optimization-first perspective naturally leads to iterative updates of the feature maps (ϕ, ψ) in function space, driven by the functional gradients of $\mathcal{L}$.

Distribution shift: detection and adaptation. The problem of distribution shift has long been recognized and continues to be an important area of research in machine learning (Quinonero-Candela et al., 2022; Moreno-Torres et al., 2012). Many existing approaches address only isolated components of the detection–adaptation pipeline, such as testing for particular types of shift (Lipton et al., 2018; Zhang et al., 2023; Goel et al., 2024), adapting uncertainty quantification wrappers (Gibbs & Candes, 2021), or treating all shifts uniformly without distinguishing harmful from benign ones (Gretton et al., 2012; Rabanser et al., 2019). In contrast, we focus on learning conditional distributions incrementally and natively under distribution shift. We show that model fit can be quantified through norms of functional gradients, and that a degradation in this quantity provides a principled signal for when adaptation is required. While several benchmarks exist for studying distribution shift, to our knowledge none are designed to evaluate the learning of conditional distributions (Cheng et al., 2025; Koh et al., 2021; Gardner et al., 2023).

3 BACKGROUND

Throughout the paper, we work with a pair of random variables (X, Y) defined on measurable spaces $(\mathcal{X}, \mathcal{F}_\mathcal{X})$ and $(\mathcal{Y}, \mathcal{F}_\mathcal{Y})$. We denote their marginal distributions by P_X and P_Y, and their joint distribution by P_{XY}. A standing assumption in our analysis is that the joint law is absolutely continuous with respect to the product measure $P_X \otimes P_Y$, that is, $P_{XY} \ll P_X \otimes P_Y$. This assumption ensures that key probabilistic objects of interest admit convenient integral representations, which will be exploited throughout the paper.

In order to study these objects from an operator-theoretic perspective, we work in spaces of square-integrable functions. Specifically, for a random variable $A \in \{X, Y\}$, we write $L^2(A)$ for the Hilbert space of functions that are square-integrable with respect to the law of A.

3.1 INTEGRAL OPERATORS AND LOW-RANK DECOMPOSITIONS

Our starting point is the conditional expectation operator, which is a canonical example of a Hilbert–Schmidt integral operator and admits a natural low-rank spectral structure. Under the standing assumption that $P_{XY} \ll P_X \otimes P_Y$, the Radon–Nikodym derivative $\kappa = dP_{XY}/d(P_X \otimes P_Y)$ is well defined and induces the conditional expectation operator $\mathsf{E} : L^2(Y) \to L^2(X)$ through the integral representation

$$[\mathsf{E}g](x) = \mathbb{E}\left[g(Y) \mid X = x\right] = \int_{\mathcal{Y}} g(y)\, \kappa(x, y)\, dP_Y(y).$$

Throughout the paper, we assume that E is compact and has finite Hilbert–Schmidt norm, a mild condition that holds for a wide class of continuous and discrete distributions; a sufficient condition is $\mathbb{E}_{P_X \otimes P_Y}[\kappa(X, Y)^2] < \infty$. Under these assumptions, the kernel κ associated with E admits a singular value expansion

$$\kappa(x, y) = 1 + \sum_{i=1}^{\infty} \sigma_i^\star\, \phi_i^\star(x)\psi_i^\star(y),$$

for some singular values $\{\sigma_i^\star\}_{i\geq 1}$ and orthonormal systems $\{\phi_i^\star\}_{i\geq 1} \subset L^2(X)$ and $\{\psi_i^\star\}_{i\geq 1} \subset L^2(Y)$. In particular, low-rank approximations of E correspond to truncations of this kernel expansion:

$$\kappa^{(d)}(x, y) = 1 + \sum_{i=1}^{d} \sigma_i^\star \phi_i^\star(x)\psi_i^\star(y),$$

Our goal is to learn from data this rank-d approximation of the conditional expectation operator, i.e., to recover the leading spectral structure of its kernel κ.

Rather than estimating the singular functions and singular values separately, we adopt a factorized parametrization that absorbs the singular values into the feature maps. Specifically, we consider models of the form

$$\kappa_{\phi,\psi}(x, y) = 1 + \langle \phi(x), \psi(y) \rangle,$$

where $\phi : \mathcal{X} \to \mathbb{R}^d$ and $\psi : \mathcal{Y} \to \mathbb{R}^d$ are learned feature maps. This parametrization is without loss of generality, as any representation of the form $\sum_{i=1}^d \sigma_i \phi_i(x)\psi_i(y)$ can be equivalently written as an inner product by rescaling the features, absorbing the singular values into ϕ and ψ.

Learning a low-rank approximation of the conditional expectation operator can be formulated as an approximation problem in the Hilbert space $L^2(P_X \otimes P_Y)$. Specifically, we measure the discrepancy between the learned kernel $\kappa_{\phi,\psi}$ and the true kernel κ using the squared $L^2(P_X \otimes P_Y)$ norm,

$$\|\kappa_{\phi,\psi} - \kappa\|^2_{L^2(P_X \otimes P_Y)} = \mathbb{E}_{P_X \otimes P_Y}\left[\left(\kappa_{\phi,\psi}(X, Y) - \kappa(X, Y)\right)^2\right].$$

Expanding the square yields

$$\|\kappa_{\phi,\psi} - \kappa\|^2_{L^2(P_X \otimes P_Y)} = \mathbb{E}_{P_X \otimes P_Y}\left[\kappa_{\phi,\psi}(X, Y)^2 - 2\kappa_{\phi,\psi}(X, Y)\, \kappa(X, Y) + \kappa(X, Y)^2\right].$$

The last term does not depend on the model parameters and can therefore be ignored for optimization purposes. To simplify the remaining expression, we use the fact that κ is the Radon–Nikodym derivative of P_{XY} with respect to $P_X \otimes P_Y$. This implies the identity

$$\mathbb{E}_{P_X \otimes P_Y}\left[\kappa_{\phi,\psi}(X, Y)\, \kappa(X, Y)\right] = \mathbb{E}_{P_{XY}}\left[\kappa_{\phi,\psi}(X, Y)\right].$$

Substituting this identity into the expansion above, we obtain that minimizing the squared $L^2(P_X \otimes P_Y)$ distance is equivalent, up to an additive constant, to minimizing the objective

$$\mathcal{L}(\phi, \psi) = -2\,\mathbb{E}_{P_{XY}}\left[\kappa_{\phi,\psi}(X, Y)\right] + \mathbb{E}_{P_X \otimes P_Y}\left[\kappa_{\phi,\psi}(X, Y)^2\right]. \tag{1}$$

Once a rank-d approximation $\kappa_{\phi,\psi}$ has been learned, the conditional CDF can be recovered via

$$F_{Y|X=x}(y) \approx \int_{-\infty}^{y} \kappa_{\phi,\psi}(x, t)\, dP_Y(t),$$

which follows from $f_{Y|X=x}(t) = \kappa(x, t)\, f_Y(t)$. In practice, this integral is estimated empirically using a held-out sample from P_Y.

3.2 Operator Learning as Functional Optimization

To learn the feature maps ϕ and ψ, we view equation 1 as an optimization problem *over functions*, with $\phi \in L^2(X)^d$ and $\psi \in L^2(Y)^d$. To formalize our approach, we briefly recall the notion of Gateaux derivatives.

A functional $F : L^2(X)^d \to \mathbb{R}$ is *Gateaux differentiable* at $\phi \in L^2(X)^d$ if there exists a bounded linear map $D_\phi F(\phi) \in \mathcal{L}(L^2(X)^d, \mathbb{R})$ such that, for every perturbation $h \in L^2(X)^d$,

$$F(\phi + \epsilon h) = F(\phi) + \epsilon\, D_\phi F(\phi)[h] + o(\epsilon),$$

where the remainder satisfies $o(\epsilon)/\epsilon \to 0$ as $\epsilon \to 0$ (Willem, 1997, Chapter 1).

By the Riesz representation theorem, there exists a unique element $\nabla_\phi F(\phi) \in L^2(X)^d$ such that

$$\nabla_\phi F(\phi)[h] = \langle h, \nabla_\phi F(\phi)\rangle_{L^2(X)^d} = \mathbb{E}_{P_X}\left[\langle h(X), \nabla_\phi F(\phi)(X)\rangle\right] \qquad \forall\, h \in L^2(X)^d.$$

We refer to $\nabla_\phi F(\phi)$ as the $L^2(X)^d$-gradient of F with respect to ϕ. An analogous definition applies to functionals of $\psi \in L^2(Y)^d$, yielding an $L^2(Y)^d$-gradient $\nabla_\psi F(\psi) \in L^2(Y)^d$.

Specializing the above definition to the objective equation 1 yields closed-form expressions for the corresponding L^2-gradients:

Proposition 3.1 (Functional derivatives of $\mathcal{L}$) *Let $\kappa_{\phi,\psi}(x, y) = 1 + \langle \phi(x), \psi(y)\rangle$, and define*

$$\mathcal{L}(\phi, \psi) = \mathbb{E}_{P_X \otimes P_Y}\left[\kappa_{\phi,\psi}(X, Y)^2\right] - 2\mathbb{E}_{P_{X,Y}}\left[\kappa_{\phi,\psi}(X, Y)\right].$$

Define the (centered) conditional-mean functions

$$m_\psi(x) := \mathbb{E}\left[\psi(Y) - \mathbb{E}_{P_Y}\left[\psi(Y)\right] \mid X = x\right] \in \mathbb{R}^d,$$
$$m_\phi(y) := \mathbb{E}\left[\phi(X) - \mathbb{E}_{P_X}\left[\phi(X)\right] \mid Y = y\right] \in \mathbb{R}^d,$$

and the (uncentered) second-moment matrices

$$C_\psi := \mathbb{E}_{P_Y}\left[\psi(Y)\psi(Y)^\top\right] \in \mathbb{R}^{d\times d}, \qquad C_\phi := \mathbb{E}_{P_X}\left[\phi(X)\phi(X)^\top\right] \in \mathbb{R}^{d\times d}.$$

Then the functional gradients of $\mathcal{L}$ are

$$\nabla_\phi \mathcal{L}(\phi, \psi)(x) = 2C_\psi\, \phi(x) - 2m_\psi(x), \qquad \nabla_\psi \mathcal{L}(\phi, \psi)(y) = 2C_\phi\, \psi(y) - 2m_\phi(y).$$

Moreover, the "diagonal" Hessian blocks are constant multiplication operators,

$$\big(\nabla^2_{\phi\phi}\mathcal{L}(\phi, \psi)\, h\big)(x) = 2C_\psi\, h(x), \qquad \big(\nabla^2_{\psi\psi}\mathcal{L}(\phi, \psi)\, g\big)(y) = 2C_\phi\, g(y),$$

for all $h \in L^2(X)^d$ and $g \in L^2(Y)^d$.

The complete calculations for Proposition 3.1 are deferred to Appendix A. In the next section, we leverage this structure to derive our training procedure: we alternate between learning these conditional-mean regressors (as standard supervised-learning subproblems) and applying a functional first- or second-order update to the feature maps.

4 A Functional Spectral-Newton Method

We now present our training method for learning a rank-d factorization of the conditional expectation operator.

4.1 NEWTON UPDATES VIA CONDITIONAL-MEAN REGRESSION

Our method is based on two observations from Proposition 3.1. First, the functional gradients are affine in the pointwise feature values:

$$\nabla_\phi \mathcal{L}(\phi,\psi)(x) = 2C_\psi\,\phi(x) - 2m_\psi(x), \qquad \nabla_\psi \mathcal{L}(\phi,\psi)(y) = 2C_\phi\,\psi(y) - 2m_\phi(y).$$

Hence, fixing ψ, the map $\phi(x) \mapsto \mathcal{L}(\phi,\psi)$ is a quadratic function for each x (and symmetrically for $\psi(y)$ when ϕ is fixed). Equivalently, the pointwise Hessians are constant multiplication operators, which makes Newton updates available in closed form.

Second, the terms m_ψ and m_ϕ are centered conditional means. In practice they are unknown and are estimated from data by treating them as regression targets; thus, each iteration interleaves (i) fitting conditional-mean regressors with (ii) applying a functional Newton step.

Concretely, at iteration t we update ψ with ϕ_t fixed, then update ϕ with ψ_{t+1} fixed:

$$\psi_{t+1}(y) = C_{\phi,t}^{-1} m_{\phi,t}(y), \qquad \phi_{t+1}(x) = C_{\psi,t+1}^{-1} m_{\psi,t+1}(x).$$

Note that the interleaved Newton update solves each block subproblem exactly in the population setting. Indeed, fix ϕ_t and view $\mathcal{L}(\phi_t,\psi)$ as a functional of ψ. By Proposition 3.1, its $L^2(Y)^d$-gradient is

$$\nabla_\psi \mathcal{L}(\phi_t,\psi)(y) = 2C_{\phi,t}\,\psi(y) - 2m_{\phi,t}(y).$$

The Newton update sets

$$\psi_{t+1}(y) = C_{\phi,t}^{-1} m_{\phi,t}(y).$$

Substituting this expression into the gradient gives, pointwise for every $y \in \mathcal{Y}$,

$$\nabla_\psi \mathcal{L}(\phi_t,\psi_{t+1})(y) = 2C_{\phi,t}\,\psi_{t+1}(y) - 2m_{\phi,t}(y) = 2C_{\phi,t}\,C_{\phi,t}^{-1} m_{\phi,t}(y) - 2m_{\phi,t}(y) = 0.$$

By symmetry, the same calculation applies to the ϕ-update, yielding $\nabla_\phi \mathcal{L}(\phi_{t+1},\psi_{t+1})(x) = 0$ for all $x \in \mathcal{X}$.

Consequently, each interleaved Newton step decreases the objective:

$$\mathcal{L}(\phi_t,\psi_{t+1}) \le \mathcal{L}(\phi_t,\psi_t), \qquad \mathcal{L}(\phi_{t+1},\psi_{t+1}) \le \mathcal{L}(\phi_t,\psi_{t+1}),$$

and therefore $\mathcal{L}(\phi_{t+1},\psi_{t+1}) \le \mathcal{L}(\phi_t,\psi_t)$ for all t.

In finite samples, we replace m_ϕ, m_ψ and C_ϕ, C_ψ by their data-driven estimates. When the conditional-mean regressors accurately approximate m_ϕ and m_ψ, and the empirical second-moment matrices provide good approximations of C_ϕ and C_ψ (so that their inverses are well behaved), the resulting updates closely track the population Newton steps above.

This highlights an important practical aspect of the method. Although the algorithm is formulated as an optimization procedure in function space, in practice it is instantiated through the estimation of the conditional-mean regressors that enter the functional gradients. Consequently, the choice of function class and training procedure used to approximate $m_{\phi,t}$ and $m_{\psi,t}$ plays a central role. Different update regimes place different demands on how accurately these conditional expectations must be learned.

In particular, Newton-type updates are inherently more "one-shot": each block step aims to solve a quadratic subproblem in a single iteration through the application of the inverse operators $C_{\phi,t}^{-1}$ and $C_{\psi,t+1}^{-1}$. As a result, effective performance in finite samples may require sufficiently expressive models and good generalization, so that both the conditional means and the second-moment operators are well approximated.

One possible modification of the above scheme is to introduce a damped (relaxed) Newton update. Instead of taking the full Newton step, we consider the scaled updates

$$\psi_{t+1} = \psi_t - \frac{\eta_\psi}{2}\, C_{\phi,t}^{-1} \nabla_\psi L(\phi_t,\psi_t), \qquad \phi_{t+1} = \phi_t - \frac{\eta_\phi}{2}\, C_{\psi,t+1}^{-1} \nabla_\phi L(\phi_t,\psi_{t+1}),$$

with relaxation parameters $\eta_\phi, \eta_\psi \in (0,1]$. Using the explicit form of the functional gradients, these updates simplify to

$$\psi_{t+1}(y) = (1-\eta_\psi)\,\psi_t(y) + \eta_\psi\, C_{\phi,t}^{-1} m_{\phi,t}(y),$$
$$\phi_{t+1}(x) = (1-\eta_\phi)\,\phi_t(x) + \eta_\phi\, C_{\psi,t+1}^{-1} m_{\psi,t+1}(x).$$

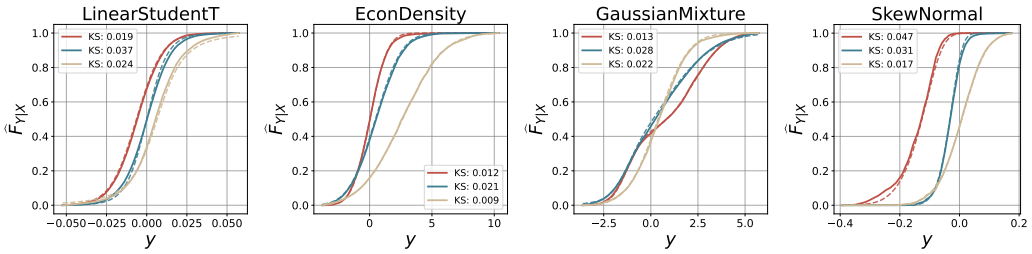

Figure 1: CDFs estimated by **FSNM** (solid) compared to the ground truth (dashed), evaluated at x equal to the 10% (red/1), 50% (blue/2), and 90% (sand/3) quantiles of X. The Kolmogorov–Smirnov (KS) statistic, reported in each legend, measures the maximum absolute deviation between the estimated and true CDFs; lower values indicate better fit.

When $\eta_\phi = \eta_\psi = 1$, we recover the pure Newton method. For smaller step sizes, the updates resemble more closely a classical gradient boosting procedure (Friedman, 2001), where each iteration adds a shrunk correction in the direction of the functional gradient rather than solving the quadratic subproblem in a single step. We do not pursue a detailed analysis of this variant here; however, we provide a concrete development of the corresponding procedure in Appendix B and include it as a natural extension of the framework.

4.2 Orthonormality Regularization

The optimization scheme above is derived for the unregularized objective $\mathcal{L}$ and highlights the central role played by the second-moment operators C_ϕ and C_ψ, which define the pointwise Hessians and appear through inverses in the Newton updates. In finite samples, however, m_ϕ and m_ψ must be learned from data and C_ϕ and C_ψ are replaced by empirical second moments. If these empirical covariances become ill-conditioned, then (i) the regressors for the conditional means may be unstable and (ii) small estimation errors in C_ϕ and C_ψ can be amplified when forming C_ϕ^{-1} and C_ψ^{-1}, causing the finite-sample updates to deviate substantially from their population counterparts. We therefore incorporate an orthonormality regularizer to control conditioning and improve numerical stability.

To this end, we encourage the learned features to be approximately orthonormal in L^2, which keeps the (empirical) covariance operators close to the identity and prevents ill conditioning in the second-order steps (Kostic et al., 2024; Fröhlich et al., 2025). Concretely, we consider the orthonormality penalty

$$\Omega(\phi, \psi) := \|C_\phi - I_d\|_{\mathrm{F}}^2 + \|C_\psi - I_d\|_{\mathrm{F}}^2 = \|C_\phi\|_{\mathrm{F}}^2 - 2\mathrm{tr}(C_\phi) + \|C_\psi\|_{\mathrm{F}}^2 - 2\mathrm{tr}(C_\psi) + 2d,$$

and, for optimization purposes, we drop the constant $2d$. Using $C_\phi = \mathbb{E}_{P_X}\left[\phi(X)\phi(X)^\top\right]$, one can show that the pointwise gradient and Hessian are

$$\nabla_{\phi(x)}\Omega_x(\phi) = 2(C_\phi - I)\phi(x), \qquad \nabla^2_{\phi(x)}\Omega_x(\phi) = 2(C_\phi - I),$$

and analogously for ψ by symmetry.

We then define the regularized objective $\mathcal{L}_\gamma(\phi, \psi) = \mathcal{L}(\phi, \psi) + \gamma\,\Omega(\phi, \psi)$, so that the regularized pointwise gradients become

$$\nabla_{\phi(x)}\mathcal{L}_{\gamma,x}(\phi, \psi) = -2\,m_\psi(x) + 2\Big(C_\psi + \gamma(C_\phi - I)\Big)\phi(x),$$
$$\nabla_{\psi(y)}\mathcal{L}_{\gamma,y}(\phi, \psi) = -2\,m_\phi(y) + 2\Big(C_\phi + \gamma(C_\psi - I)\Big)\psi(y).$$

5 Harmful Drift Detection via Functional Gradient Norms

Our goal is to detect whether the test distribution Q_{XY} has drifted from the train distribution P_{XY}.

Let $(\tilde{X}, \tilde{Y}) \sim Q_{XY}$ with marginals Q_X and Q_Y, which may differ from the training distribution P_{XY} (and its marginals). Assume $Q_X \ll P_X$, and fix a (near-)stationary iterate $(\phi^\star, \psi^\star)$ of the

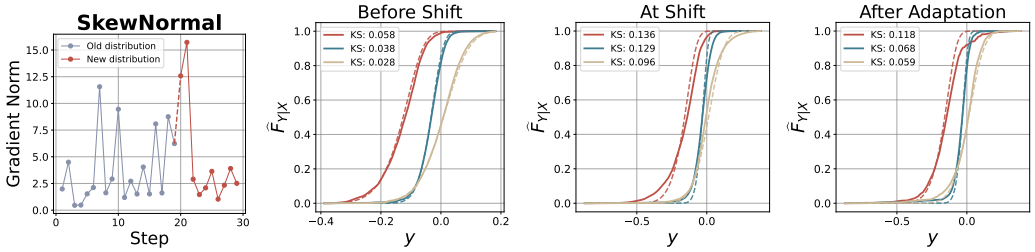

Figure 2: Performance of our method (**FSNM**) under covariate drift.

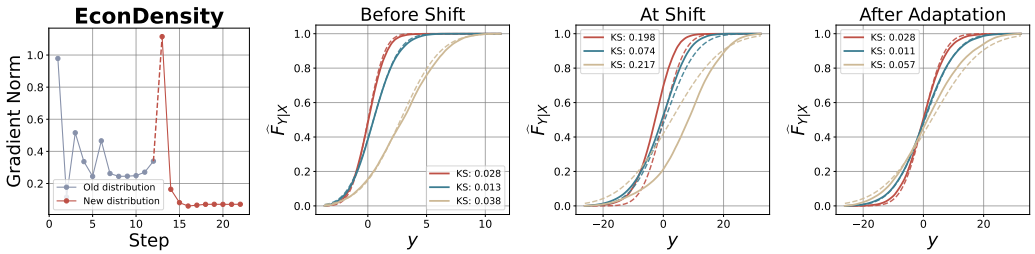

Figure 3: Performance of our method (**FSNM**) under concept drift.

training procedure. Define the per-sample ϕ-gradient at this iterate by

$$F(x) := \nabla_\phi \mathcal{L}(\phi^\star, \psi^\star)(x).$$

Given a test batch $\tilde{S}_M = \{\tilde{X}_i\}_{i=1}^M$ with $\tilde{X}_i \sim Q$ and a calibration sample $S_N = \{X_j\}_{j=1}^N$ with $X_j \sim P$, we consider the mean-gradient statistic

$$T(\tilde{S}_M) := \left\| \frac{1}{M} \sum_{i=1}^{M} F(\tilde{X}_i) \right\|^2 .$$

Under the null hypothesis $H_0 : Q_{XY} = P_{XY}$, the pooled sample $(\tilde{X}_{1:M}, X_{1:N})$ is exchangeable. Thus, the labels "test" versus "calibration" are arbitrary. This enables us to calibrate $T(\tilde{S}_M)$ without concentration assumptions via a permutation test: let $Z_{1:M+N} = (\tilde{X}_1, \ldots, \tilde{X}_M, X_1, \ldots, X_N)$. For $b = 1, \ldots, B$, draw a subset $\mathcal{I}_b \subset \{1, \ldots, M+N\}$ uniformly at random among all subsets of size M, define $\tilde{S}_M^{(b)} = \{Z_i : i \in \mathcal{I}_b\}$, and compute $T_b := T(\tilde{S}_M^{(b)})$. The permutation p-value is then

$$p = \frac{1 + \#\{b : T_b \geq T(\tilde{S}_M)\}}{B+1}.$$

Rejecting H_0 when $p \leq \alpha$ yields a finite-sample level-α test under exchangeability.

In this work, we focus on evaluating the mean-gradient statistic itself, without performing a formal test—analogous to computing R^2 in linear regression without testing for goodness-of-fit. We therefore characterize harmful distribution shifts as those that induce large norms of the functional gradients. Since the gradient vanishes at a well-fitted model, a large $\|F\|$ under new data directly measures how much the conditional structure has degraded—and adaptation follows naturally by continuing the Newton updates.

6 Experiments

In this section, we report the results of numerical experiments for the task of conditional distribution estimation under distribution shifts. All experiments are conducted using the data-generating models from the established conditional density estimation benchmark[1] of Rothfuss et al. (2019), with

[1]Benchmark code is available at `github.com/freelunchtheorem/Conditional_Density_Estimation`.

conditional means estimated via multivariate gradient boosting using XGBoost (Chen & Guestrin, 2016). Full experiments details are given in Appendix C.

Baseline without shifts. We first evaluate the performance of our proposed method (**FSNM**) in a setting without distributional shifts. We consider the following data models from Rothfuss et al. (2019):

- `LinearStudentT`: a conditional Student-t distribution defined as
$$Y \mid X \sim \text{Student-}t\big(\mu(X), \sigma(X), \nu(X)\big), \qquad X \sim \mathcal{N}(0,1),$$
with $\nu(x) = \nu_{\text{slope}}\,\text{sigmoid}(-2x)+\nu_{\text{cst}}$; $\mu(x) = \mu_{\text{slope}}x+\mu_{\text{cst}}$; and $\sigma(x) = \sigma_{\text{slope}}|x|+\sigma_{\text{cst}}$.
- `EconDensity`: an economically inspired heteroscedastic density model with a quadratic dependence on the conditional variable, defined as
$$Y = X^2 + \varepsilon_Y, \qquad \varepsilon_Y \sim \mathcal{N}\left(0, \alpha^2(1+X)^2\right), \qquad X \sim |\mathcal{N}(0,1)|.$$
- `GaussianMixture`: a bivariate Gaussian mixture model with five components, defined as
$$f_{XY}(X,Y) = \sum_{k=1}^{5} \pi_k \mathcal{N}(\mu_k, \Sigma_k),$$
where π_k, μ_k, and Σ_k denote the mixing coefficients, mean vectors, and covariance matrices.
- `SkewNormal`: a univariate skew-normal distribution defined as
$$Y = 2\,\phi(X)\,\psi(\alpha X),$$
where $\phi(\cdot)$ and $\psi(\cdot)$ are the standard normal density and cumulative distribution functions.

The results are shown in Figure 1 for x equal to the 10%, 50%, and 90% quantiles of X. We can see that **FSNM** is able to correctly estimate the conditional distribution functions (CDFs) with low Kolmogorov-Smirnov (KS) values ($< 10^{-1}$).

Covariate shift. We begin by exploring covariate shift on the `SkewNormal` model by altering the marginal distribution of X from $\mathcal{N}(0, 0.25)$ to $\text{Laplace}(0, 0.25)$.

The results are shown in Figure 2, where x corresponds to the 10%, 50%, and 90% quantiles of X. We observe a clear spike in the norm of the functional gradient of ϕ at step $T = 20$, indicating a degradation of the fit. This behavior is also reflected in Figure 2 (third panel from the left), which shows both a qualitative deterioration of the estimated CDFs and an increase in KS values. After adaptation (Figure 2, right), the predictive performance is partially restored. The recovery is only partial because the Laplace distribution has heavier tails, making it inherently more challenging to estimate.

Concept shift. We further evaluate our approach on concept drift on the `EconDensity` model by altering the noise standard deviation $\sigma_{\varepsilon_Y} = \alpha(1+X)$ from $\alpha = 1$ during train to $\alpha = 5$ during test.

The results are shown in Figure 3. We observe a pronounced spike in the norms of the functional gradients at step $T = 13$, signaling a degradation in model fit (Figure 3, left). This degradation is also evident in the learned conditional distribution functions (Figure 3, second and third panels from left). To address this, we adapt the model by following the functional gradients, which restores the quality of the learned CDFs (Figure 3, right).

7 Conclusions

In this paper, we presented a stagewise framework for learning conditional distributions that is explicitly designed to operate under distribution shift. By learning a spectral decomposition of the density ratio through alternating functional Newton updates, our method natively supports incremental learning and targeted adaptation. Central to this framework is a statistically grounded criterion, based on norms of functional gradients, for identifying shifts that are harmful to predictive performance and warrant intervention. Empirical results on conditional density estimation benchmarks with induced shifts show that the proposed approach reliably detects harmful drifts and enables effective performance recovery. Together, these results suggest a principled path toward robust conditional modeling in high-risk, nonstationary environments.

REFERENCES

Anastasios N. Angelopoulos and Stephen Bates. Conformal prediction: A gentle introduction. *Foundations and Trends in Machine Learning*, 16(4):494–591, March 2023. ISSN 1935-8245. doi: 10.1561/2200000101. URL http://dx.doi.org/10.1561/2200000101.

Peter Bühlmann and Torsten Hothorn. Boosting algorithms: Regularization, prediction and model fitting. *Statistical Science*, 22(4), November 2007. ISSN 0883-4237. doi: 10.1214/07-sts242. URL http://dx.doi.org/10.1214/07-STS242.

Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, pp. 785–794. ACM, August 2016. doi: 10.1145/2939672.2939785. URL http://dx.doi.org/10.1145/2939672.2939785.

Zi-Jian Cheng, Zi-Yi Jia, Zhi Zhou, Yu-Feng Li, and Lan-Zhe Guo. Tabfsbench: Tabular benchmark for feature shifts in open environments, 2025. URL https://arxiv.org/abs/2501.18935.

Victor Chernozhukov, Kaspar Wüthrich, and Yinchu Zhu. Distributional conformal prediction. *Proceedings of the National Academy of Sciences*, 118(48), November 2021. ISSN 1091-6490. doi: 10.1073/pnas.2107794118. URL http://dx.doi.org/10.1073/pnas.2107794118.

Daniel Csillag, Lucas Monteiro Paes, Thiago Ramos, João Vitor Romano, Rodrigo Schuller, Roberto B. Seixas, Roberto I. Oliveira, and Paulo Orenstein. Amnioml: Amniotic fluid segmentation and volume prediction with uncertainty quantification. *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(13):15494–15502, June 2023. ISSN 2159-5399. doi: 10.1609/aaai.v37i13.26837. URL http://dx.doi.org/10.1609/aaai.v37i13.26837.

Sam Efromovich. *Nonparametric curve estimation*. Springer, 1999.

Samuel G. Finlayson, Adarsh Subbaswamy, Karandeep Singh, John Bowers, Annabel Kupke, Jonathan Zittrain, Isaac S. Kohane, and Suchi Saria. The clinician and dataset shift in artificial intelligence. *New England Journal of Medicine*, 385(3):283–286, July 2021. ISSN 1533-4406. doi: 10.1056/nejmc2104626. URL http://dx.doi.org/10.1056/NEJMc2104626.

Jerome H Friedman. Greedy function approximation: a gradient boosting machine. *Annals of statistics*, pp. 1189–1232, 2001.

Alek Fröhlich, Thiago Ramos, Gustavo Motta Cabello Dos Santos, Isabela Panzeri Carlotti Buzatto, Rafael Izbicki, and Daniel Guimarães Tiezzi. Personalizedus: Interpretable breast cancer risk assessment with local coverage uncertainty quantification. *Proceedings of the AAAI Conference on Artificial Intelligence*, 39(27):27998–28006, April 2025. ISSN 2159-5399. doi: 10.1609/aaai.v39i27.35017. URL http://dx.doi.org/10.1609/aaai.v39i27.35017.

Alek Fröhlich, Vladimir Kostic, Karim Lounici, Daniel Perazzo, and Massimiliano Pontil. Toward scalable and valid conditional independence testing with spectral representations, 2025. URL https://arxiv.org/abs/2512.19510.

Josh Gardner, Zoran Popovic, and Ludwig Schmidt. Benchmarking distribution shift in tabular data with tableshift. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 53385–53432. Curran Associates, Inc., 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/a76a757ed479a1e6a5f8134bea492f83-Paper-Datasets_and_Benchmarks.pdf.

Isaac Gibbs and Emmanuel Candes. Adaptive conformal inference under distribution shift. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, 2021. URL https://openreview.net/forum?id=6vaActvpcp3.

Surbhi Goel, Abhishek Shetty, Konstantinos Stavropoulos, and Arsen Vasilyan. Tolerant algorithms for learning with arbitrary covariate shift. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 124979–125018. Curran Associates, Inc., 2024. doi: 10.52202/079017-3969. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/e209210eae282e23e305df49fbb2769c-Paper-Conference.pdf.

Arthur Gretton, Karsten M. Borgwardt, Malte J. Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. *Journal of Machine Learning Research*, 13(25):723–773, 2012. URL http://jmlr.org/papers/v13/gretton12a.html.

Alexander Grubb and J. Andrew Bagnell. Generalized boosting algorithms for convex optimization, 2012. URL https://arxiv.org/abs/1105.2054.

Jeff Z. HaoChen, Colin Wei, Adrien Gaidon, and Tengyu Ma. Provable guarantees for self-supervised deep learning with spectral contrastive loss. In *Advances in Neural Information Processing Systems*, volume 34, pp. 5000–5011, 2021.

Yang Hu, Tianyi Chen, Na Li, Kai Wang, and Bo Dai. Primal-dual spectral representation for off-policy evaluation. *arXiv preprint arXiv:2410.17538*, 2024.

Rafael Izbicki. *Spectral Series Approach to High-Dimensional Nonparametric Inference*. PhD thesis, Carnegie Mellon University, 2014.

Pang Wei Koh, Shiori Sagawa, Henrik Marklund, Sang Michael Xie, Marvin Zhang, Akshay Balsubramani, Weihua Hu, Michihiro Yasunaga, Richard Lanas Phillips, Irena Gao, Tony Lee, Etienne David, Ian Stavness, Wei Guo, Berton Earnshaw, Imran Haque, Sara M Beery, Jure Leskovec, Anshul Kundaje, Emma Pierson, Sergey Levine, Chelsea Finn, and Percy Liang. Wilds: A benchmark of in-the-wild distribution shifts. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 5637–5664. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/koh21a.html.

Vladimir R. Kostic, Karim Lounici, Grégoire Pacreau, Giacomo Turri, Pietro Novelli, and Massimiliano Pontil. Neural conditional probability for uncertainty quantification. In *Advances in Neural Information Processing Systems*, volume 37, pp. 60999–61039, 2024.

Zachary Lipton, Yu-Xiang Wang, and Alexander Smola. Detecting and correcting for label shift with black box predictors. In Jennifer Dy and Andreas Krause (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 3122–3130. PMLR, 10–15 Jul 2018. URL https://proceedings.mlr.press/v80/lipton18a.html.

David G. Luenberger. *Optimization by vector space methods*. John Wiley & Sons Inc., New York, 1969.

Llew Mason, Jonathan Baxter, Peter Bartlett, and Marcus Frean. Boosting algorithms as gradient descent. In S. Solla, T. Leen, and K. Müller (eds.), *Advances in Neural Information Processing Systems*, volume 12. MIT Press, 1999. URL https://proceedings.neurips.cc/paper_files/paper/1999/file/96a93ba89a5b5c6c226e49b88973f46e-Paper.pdf.

Dimitri Meunier, Antoine Moulin, Jakub Wornbard, Vladimir R. Kostic, and Arthur Gretton. Demystifying spectral feature learning for instrumental variable regression. *arXiv preprint arXiv:2506.10899*, 2025a.

Dimitri Meunier, Jakub Wornbard, Vladimir R. Kostic, Antoine Moulin, Alek Fröhlich, Karim Lounici, Massimiliano Pontil, and Arthur Gretton. Outcome-aware spectral feature learning for instrumental variable regression. *arXiv preprint arXiv:2512.00919*, 2025b.

Jose G. Moreno-Torres, Troy Raeder, Rocío Alaiz-Rodríguez, Nitesh V. Chawla, and Francisco Herrera. A unifying view on dataset shift in classification. *Pattern Recognition*, 45(1):521–530, January 2012. ISSN 0031-3203. doi: 10.1016/j.patcog.2011.06.019. URL http://dx.doi.org/10.1016/j.patcog.2011.06.019.

Daniel Ordoñez-Apraez, Vladimir Kostić, Alek Fröhlich, Vivien Brandt, Karim Lounici, and Massimiliano Pontil. Equivariant representation learning for symmetry-aware inference with guarantees. *arXiv preprint arXiv:2505.19809*, 2025.

Joaquin Quinonero-Candela, Masashi Sugiyama, Anton Schwaighofer, and Neil D Lawrence (eds.). *Dataset shift in machine learning*. MIT Press, 2022.

Stephan Rabanser, Stephan Günnemann, and Zachary Lipton. Failing loudly: An empirical study of methods for detecting dataset shift. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/file/846c260d715e5b854ffad5f70a516c88-Paper.pdf.

Jonas Rothfuss, Fabio Ferreira, Simon Walther, and Maxim Ulrich. Conditional density estimation with neural networks: Best practices and benchmarks. *arXiv:1903.00954*, 2019.

Robert E. Schapire and Yoav Freund. *Boosting: Foundations and Algorithms*. The MIT Press, 05 2012. ISBN 9780262301183. doi: 10.7551/mitpress/8291.001.0001. URL https://doi.org/10.7551/mitpress/8291.001.0001.

Ewout W. Steyerberg. *Clinical Prediction Models: A Practical Approach to Development, Validation, and Updating*. Springer International Publishing, 2019. ISBN 9783030163990. doi: 10.1007/978-3-030-16399-0. URL http://dx.doi.org/10.1007/978-3-030-16399-0.

Masashi Sugiyama, Taiji Suzuki, and Takafumi Kanamori. *Density Ratio Estimation in Machine Learning*. Cambridge University Press, 2012.

Haotian Sun, Antoine Moulin, Tongzheng Ren, Arthur Gretton, and Bo Dai. Spectral representation for causal estimation with hidden confounders. In *The 28th International Conference on Artificial Intelligence and Statistics*, pp. 2719–2727. PMLR, 2025.

Giacomo Turri, Luigi Bonati, Kai Zhu, Massimiliano Pontil, and Pietro Novelli. Self-supervised evolution operator learning for high-dimensional dynamical systems. *arXiv preprint arXiv:2505.18671*, 2025.

Ben Van Calster, Ewout W. Steyerberg, Laure Wynants, and Maarten van Smeden. There is no such thing as a validated prediction model. *BMC Medicine*, 21(1), February 2023. ISSN 1741-7015. doi: 10.1186/s12916-023-02779-w. URL http://dx.doi.org/10.1186/s12916-023-02779-w.

Ziyu Wang, Yucen Luo, Yueru Li, Jun Zhu, and Bernhard Schölkopf. Spectral representation learning for conditional moment models. *arXiv preprint arXiv:2210.16525*, 2022.

M. Willem. *Minimax Theorems*. Progress in Nonlinear Differential Equations and Their Applications. Birkhäuser Boston, 1997. ISBN 9783764339135. URL https://books.google.com.br/books?id=Q8fvAAAAMAAJ.

Yu-Jie Zhang, Zhen-Yu Zhang, Peng Zhao, and Masashi Sugiyama. Adapting to continuous covariate shift via online density ratio estimation. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 29074–29113. Curran Associates, Inc., 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/5cad96c4433955a2e76749ee74a424f5-Paper-Conference.pdf.

A FUNCTIONAL DERIVATIVES CALCULATIONS

In this section we compute the functional gradients and Hessian of our loss $\mathcal{L}(\phi, \psi)$.

A.1 FIRST-ORDER DERIVATIVES

Gradient of product expectation. Under the product measure $P_X \otimes P_Y$, we have

$$\begin{aligned}
\mathbb{E}_{P_X \otimes P_Y}\left[\kappa_{\phi,\psi}(X,Y)^2\right] &= \mathbb{E}_{P_X \otimes P_Y}\left[(1 + \langle \phi(X), \psi(Y)\rangle)^2\right] \\
&= 1 + \mathbb{E}_{P_X \otimes P_Y}\left[\langle \phi(X), \psi(Y)\rangle^2\right] + 2\mathbb{E}_{P_X \otimes P_Y}\left[\langle \phi(X), \psi(Y)\rangle\right] \\
&= 1 + \mathbb{E}_{P_X \otimes P_Y}\left[\phi(X)^\top \psi(Y)\psi(Y)^\top \phi(X)\right] + 2\langle \mathbb{E}_{P_X}\left[\phi(X)\right], \mathbb{E}_{P_Y}\left[\psi(Y)\right]\rangle \\
&= 1 + \mathbb{E}_{P_X \otimes P_Y}\left[\mathrm{tr}(\phi(X)\phi(X)^\top \psi(Y)\psi(Y)^\top)\right] + 2\langle \mathbb{E}_{P_X}\left[\phi(X)\right], \mathbb{E}_{P_Y}\left[\psi(Y)\right]\rangle \\
&= 1 + \mathrm{tr}\left(\mathbb{E}_{P_X \otimes P_Y}\left[\phi(X)\phi(X)^\top \psi(Y)\psi(Y)^\top\right]\right) + 2\langle \mathbb{E}_{P_X}\left[\phi(X)\right], \mathbb{E}_{P_Y}\left[\psi(Y)\right]\rangle \\
&= 1 + \mathrm{tr}\left(\mathbb{E}_{P_X}\left[\phi(X)\phi(X)^\top\right]\mathbb{E}_{P_Y}\left[\psi(Y)\psi(Y)^\top\right]\right) + 2\langle \mathbb{E}_{P_X}\left[\phi(X)\right], \mathbb{E}_{P_Y}\left[\psi(Y)\right]\rangle,
\end{aligned}$$

where the last step uses the independence of $X \sim P_X$ and $Y \sim P_Y$ under $P_X \otimes P_Y$.

Moreover,

$$\begin{aligned}
\mathrm{tr}\big(\mathbb{E}_{P_X}\left[\phi(X)\phi(X)^\top\right]\mathbb{E}_{P_Y}\left[\psi(Y)\psi(Y)^\top\right]\big) &= \mathrm{tr}\big(\mathbb{E}_{P_X}\left[\phi(X)\phi(X)^\top C_\psi\right]\big) \\
&= \mathbb{E}_{P_X}\left[\mathrm{tr}\left(\phi(X)\phi(X)^\top C_\psi\right)\right] \\
&= \mathbb{E}_{P_X}\left[\phi(X)^\top C_\psi\, \phi(X)\right].
\end{aligned}$$

The product term is quadratic in ϕ and ψ, and its Gateaux derivatives admit closed forms. For any perturbation h of ϕ,

$$\nabla_\phi\left(\mathbb{E}_{P_X \otimes P_Y}\left[\kappa_{\phi,\psi}(X,Y)^2\right]\right)[h] = 2\mathbb{E}_{P_X}\left[\langle h(X),\ C_\psi\, \phi(X) + \mathbb{E}_{P_Y}\left[\psi(Y)\right]\rangle\right].$$

Similarly, for any perturbation g of ψ,

$$\nabla_\psi\left(\mathbb{E}_{P_X \otimes P_Y}\left[\kappa_{\phi,\psi}(X,Y)^2\right]\right)[g] = 2\mathbb{E}_{P_Y}\left[\langle g(Y),\ C_\phi\, \psi(Y) + \mathbb{E}_{P_X}\left[\phi(X)\right]\rangle\right].$$

Gradient of joint expectation. The Gateaux derivative with respect to ϕ is

$$\nabla_\phi\left(\mathbb{E}_{P_{X,Y}}\left[\kappa_{\phi,\psi}(X,Y)\right]\right)[h] = \mathbb{E}_{P_{X,Y}}\left[\langle h(X), \psi(Y)\rangle\right].$$

To express the directional derivative in Gateaux form—that is, as an $L^2(X)^d$ inner product against h—we apply the tower property of conditional expectations:

$$\begin{aligned}
\mathbb{E}_{P_{X,Y}}\left[\langle h(X), \psi(Y)\rangle\right] &= \mathbb{E}_{P_X}\left[\mathbb{E}\left[\langle h(X), \psi(Y)\rangle \mid X\right]\right] \\
&= \mathbb{E}_{P_X}\left[\langle h(X), \mathbb{E}\left[\psi(Y) \mid X\right]\rangle\right].
\end{aligned}$$

This shows that the $L^2(X)^d$-gradient of the joint term with respect to ϕ is the function $x \mapsto \mathbb{E}\left[\psi(Y) \mid X = x\right]$.

Similarly, the $L^2(Y)^d$-gradient with respect to ψ is the function $y \mapsto \mathbb{E}\left[\phi(X) \mid Y = y\right]$.

Putting everything together, the L^2 functional gradients are

$$\begin{aligned}
\nabla_\phi \mathcal{L}(\phi, \psi) &= 2C_\psi\, \phi - 2m_\psi, \\
\nabla_\psi \mathcal{L}(\phi, \psi) &= 2C_\phi\, \psi - 2m_\phi.
\end{aligned}$$

A.2 SECOND-ORDER DERIVATIVES

By definitoin, the $\phi\phi$-block of the Hessian is the bounded bilinear map

$$\nabla_\phi \mathcal{L}(\phi + \varepsilon\tilde{h}, \psi)[h] = \nabla_\phi \mathcal{L}(\phi, \psi)[h] + \varepsilon\, \nabla^2_{\phi\phi}\mathcal{L}(\phi, \psi)[\tilde{h}, h] + o(\varepsilon), \qquad \forall\, h, \tilde{h} \in L^2(X)^d,$$

and similarly for $\nabla^2_{\psi\psi}\mathcal{L}(\phi, \psi)$ on $L^2(Y)^d$.

From the L^2-gradient expression

$$\nabla_\phi \mathcal{L}(\phi, \psi)(x) = 2C_\psi\, \phi(x) - 2m_\psi(x),$$

and using that m_ψ does not depend on ϕ, we have for any $h \in L^2(X)^d$

$$\nabla_\phi \mathcal{L}(\phi + \varepsilon h, \psi)(x) = \nabla_\phi \mathcal{L}(\phi, \psi)(x) + \varepsilon\, 2C_\psi\, h(x).$$

Therefore, for any $\tilde{h}, h \in L^2(X)^d$,

$$\begin{aligned}
\nabla^2_{\phi\phi} \mathcal{L}(\phi, \psi)[\tilde{h}, h] &= \lim_{\varepsilon \to 0} \frac{\nabla_\phi \mathcal{L}(\phi + \varepsilon\tilde{h}, \psi)[h] - \nabla_\phi \mathcal{L}(\phi, \psi)[h]}{\varepsilon} \\
&= \lim_{\varepsilon \to 0} \frac{\mathbb{E}_{P_X}\left[\langle h(X),\, \nabla_\phi \mathcal{L}(\phi + \varepsilon\tilde{h}, \psi)(X)\rangle\right] - \mathbb{E}_{P_X}\left[\langle h(X),\, \nabla_\phi \mathcal{L}(\phi, \psi)(X)\rangle\right]}{\varepsilon} \\
&= \mathbb{E}_{P_X}\left[\langle h(X),\, 2C_\psi\, \tilde{h}(X)\rangle\right] = \langle h,\, 2C_\psi\, \tilde{h}\rangle_{L^2(X)^d}.
\end{aligned}$$

Equivalently, identifying the bilinear form with its Riesz operator representation on $L^2(X)^d$, the $\phi\phi$-Hessian acts as

$$\big(\nabla^2_{\phi\phi} \mathcal{L}(\phi, \psi)\, h\big)(x) = 2C_\psi\, h(x), \qquad \forall\, h \in L^2(X)^d,$$

so it is constant (independent of (ϕ, ψ)) and equal to the multiplication operator by $2C_\psi$.

An analogous argument starting from

$$\nabla_\psi \mathcal{L}(\phi, \psi)(y) = 2C_\phi\, \psi(y) - 2m_\phi(y)$$

yields, for any $\tilde{g}, g \in L^2(Y)^d$,

$$\nabla^2_{\psi\psi} \mathcal{L}(\phi, \psi)[\tilde{g}, g] = \mathbb{E}_{P_Y}\left[\langle g(Y),\, 2C_\phi\, \tilde{g}(Y)\rangle\right] = \langle g,\, 2C_\phi\, \tilde{g}\rangle_{L^2(Y)^d},$$

and the corresponding operator form

$$\big(\nabla^2_{\psi\psi} \mathcal{L}(\phi, \psi)\, g\big)(y) = 2C_\phi\, g(y), \qquad \forall\, g \in L^2(Y)^d.$$

For the crossed terms, let $g \in L^2(Y)^d$ be a perturbation direction (i.e., $\psi \mapsto \psi + \epsilon g$). Expanding the second-moment matrix yields

$$\begin{aligned}
C_\psi(\psi + \epsilon g) &= \mathbb{E}_{P_Y}\left[(\psi(Y) + \epsilon g(Y))(\psi(Y) + \epsilon g(Y))^\top\right] \\
&= C_\psi(\psi) + \epsilon\, \mathbb{E}_{P_Y}\left[g(Y)\psi(Y)^\top + \psi(Y)g(Y)^\top\right] + O(\epsilon^2),
\end{aligned}$$

and hence

$$\nabla_\psi C_\psi[g] = \mathbb{E}_{P_Y}\left[g(Y)\psi(Y)^\top + \psi(Y)g(Y)^\top\right].$$

Next, consider the centered conditional mean

$$m_\psi(x) = \mathbb{E}\left[\psi(Y) - \mathbb{E}_{P_Y}\left[\psi(Y)\right] \mid X = x\right] = \mathbb{E}\left[\psi(Y) \mid X = x\right] - \mathbb{E}_{P_Y}\left[\psi(Y)\right].$$

By linearity of conditional expectation and of the marginal expectation,

$$\nabla_\psi m_\psi(x)[g] = \mathbb{E}\left[g(Y) \mid X = x\right] - \mathbb{E}_{P_Y}\left[g(Y)\right].$$

Combining these identities with $\nabla_\phi \mathcal{L}(\phi, \psi)(x) = 2C_\psi\, \phi(x) - 2m_\psi(x)$, we obtain the crossed Hessian block $\nabla^2_{\phi\psi} \mathcal{L}(\phi, \psi)$ as

$$\big(\nabla^2_{\phi\psi} \mathcal{L}(\phi, \psi)\, g\big)(x) = 2\, \nabla_\psi C_\psi[g]\; \phi(x) - 2\, \nabla_\psi m_\psi(x)[g].$$

Equivalently, the associated bilinear form is obtained by pairing the image $\nabla^2_{\phi\psi} \mathcal{L}(\phi, \psi)\, g \in L^2(X)^d$ against any $\tilde{h} \in L^2(X)^d$ via

$$\nabla^2_{\phi\psi} \mathcal{L}(\phi, \psi)[\tilde{h}, g] := \langle \tilde{h},\; \nabla^2_{\phi\psi} \mathcal{L}(\phi, \psi)\, g\rangle_{L^2(X)^d} = \mathbb{E}_{P_X}\left[\left\langle \tilde{h}(X),\; (\nabla^2_{\phi\psi} \mathcal{L}(\phi, \psi)\, g)(X)\right\rangle\right].$$

Similarly, we obtain the crossed block $\nabla^2_{\psi\phi} \mathcal{L}(\phi, \psi)$:

$$\big(\nabla^2_{\psi\phi} \mathcal{L}(\phi, \psi)\, h\big)(y) = 2\, \nabla_\phi C_\phi[h]\; \psi(y) - 2\, \nabla_\phi m_\phi(y)[h],$$

with $\nabla_\phi C_\phi[h] = \mathbb{E}_{P_X}\left[h(X)\phi(X)^\top + \phi(X)h(X)^\top\right]$, and an implicit bilinear representation obtained by pairing against any $\tilde{g} \in L^2(Y)^d$ in the $L^2(Y)^d$ inner product.

B Relaxed Newton Method

In this appendix we detail a relaxed version of the functional Newton procedure. The goal is to interpolate between the pure Newton method, which solves each quadratic block subproblem in a single step, and a more conservative functional gradient-descent regime.

To this end, we introduce relaxation parameters $\eta_\phi, \eta_\psi \in (0, 1]$ and replace the exact Newton step by a convex combination between the current iterate and the Newton solution. Implemented in an interleaved (alternating) manner, the updates become

$$\psi_{t+1}(y) = (1 - \eta_\psi)\, \psi_t(y) + \eta_\psi\, C_{\phi,t}^{-1} m_{\phi,t}(y), \tag{2}$$

$$\phi_{t+1}(x) = (1 - \eta_\phi)\, \phi_t(x) + \eta_\phi\, C_{\psi,t+1}^{-1} m_{\psi,t+1}(x). \tag{3}$$

When $\eta_\phi = \eta_\psi = 1$, we recover the pure Newton method. For smaller step sizes, each block update moves only partially toward the Newton solution, resulting in a more gradual descent.

Unrolling equation 3–equation 2 gives the explicit representations

$$\psi_t(y) = (1 - \eta_\psi)^t \psi_0(y) + \eta_\psi \sum_{k=0}^{t-1} (1 - \eta_\psi)^{t-1-k}\, C_{\phi,k}^{-1} m_{\phi,k}(y),$$

$$\phi_t(x) = (1 - \eta_\phi)^t \phi_0(x) + \eta_\phi \sum_{k=0}^{t-1} (1 - \eta_\phi)^{t-1-k}\, C_{\psi,k}^{-1} m_{\psi,k}(x).$$

These expressions make explicit that the relaxed iterates form exponentially weighted averages of past Newton directions. In particular, the contribution of earlier updates decays geometrically at rate $(1 - \eta_\phi)$ and $(1 - \eta_\psi)$, respectively.

From a statistical perspective, the damping reduces the sensitivity of the method to estimation error in $m_{\phi,t}, m_{\psi,t}$ and in the empirical operators $C_{\phi,t}, C_{\psi,t}$. Because each step applies only a fraction of the Newton correction, approximation errors are less aggressively propagated through the recursion. This makes it possible to employ simpler function classes for the conditional-mean regressors, at the cost of requiring more iterations. A precise characterization can be obtained by explicitly tracking how regression error propagates through the relaxed recursion.

To formalize this intuition, one may impose an *edge* condition, analogous to the weak-learner assumption in boosting. Let $\nabla_\phi L(\phi_t, \psi_t)$ denote the population functional gradient, and let $\nabla_\phi \widehat{L}_t(\phi_t, \psi_t)$ denote its sample-based approximation obtained from the learned conditional-mean regressors at iteration t. Assume there exists $\gamma \in (0, 1)$ such that

$$\left\| \nabla_\phi L(\phi_t, \psi_t) - \nabla_\phi \widehat{L}_t(\phi_t, \psi_t) \right\|_{L^2(X)^d}^2 \le (1 - \gamma^2) \left\| \nabla_\phi L(\phi_t, \psi_t) \right\|_{L^2(X)^d}^2.$$

Intuitively, this condition ensures that the learned update direction retains a fixed fraction of alignment with the true functional gradient.

Under such an edge assumption, the relaxed recursion inherits a geometric contraction property similar to that of classical gradient boosting. In particular, the population gradient decreases at a geometric rate under the relaxed step, and the approximation error can be shown to track the population dynamics. Consequently, as in the boosting analysis of Grubb & Bagnell (2012), one obtains convergence of the relaxed iterates provided the step size is chosen so that the underlying contraction factor remains strictly below one.

C Experiments Details

In this section, we will give an overview of the experimental set-up used in this paper.

Implementation details. We employed XGBoost with its built-in early stopping and further implemented early stopping at the overall training level. To enhance numerical stability, the model was initialized using an orthonormal sine basis. At each iteration, gradient steps were computed through

a least-squares formulation, further improving stability. Updates to the function parameters were carried out via an alternating optimization scheme. For the benchmark experiments, we used 1000 boosting rounds with a squared-error objective in XGBoost.

Hyperparameter optimization. Our method (**FSNM**) was tuned using Weights & Biases with the selection `method` set to `bayes`. We performed 100 optimization trials on the hyperparameter grid reported in Table 1. The selection of the hyperparameters was based on minimizing the following validation score:

$$S_{\text{val}} = \frac{1}{5}\sum_{i=1}^{5} \text{KS}(F_{Y|X=x_i}, \widehat{F}_{Y|X=x_i}),$$

where x_i denote the $10, 25, 50, 75, 90\%$ quantiles of X and KS denotes the Kolmogorov-Smirnov metric:

$$\text{KS}(F_{Y|X=x}, \widehat{F}_{Y|X=x}) = \sup_{y\in\mathbb{R}} \left| F_{Y|X=x}(y) - \widehat{F}_{Y|X=x}(y) \right|. \tag{4}$$

Table 1: Hyperparameter grid for benchmark.

Hyperparameter	**Values**	**Description**
`gamma`	$[10^{-4},\ 3\times10^{-2}]$	Regularization γ
`learning_rate_phi`	$[2\times10^{-4},\ 2\times10^{-1}]$	Learning rate for ϕ updates
`learning_rate_psi`	$[2\times10^{-4},\ 2\times10^{-1}]$	Learning rate for ψ updates
`reg_alpha_{phi,psi}`	$[10^{-4},\ 2\times10^{-2}]$	Regularization strength for tree models
`n_estimators_{phi,psi}`	$\phi : [10, 300]$, $\psi : [10, 1000]$	Number of boosting trees
`max_depth_{phi,psi}`	$\{1, 3, 5, 7, 9\}$	Maximum depth of each tree
`subsample_{phi,psi}`	$\{0.5, 0.7, 0.9\}$	Row subsampling ratio during training
`colsample_bytree_{phi,psi}`	$\{0.7, 0.9, 0.95\}$	Column subsampling ratio per tree
`D`	$\{200, 250, 300, 350, 400\}$	Dimension / number of basis functions
`init_random`	{random_trees, orthonormal_sines}	Initialization strategy for basis functions

For the chosen hyperparameters in the benchmark, we used the values listed in Table 2.

Table 2: Hyperparameters used for different benchmark experiments.

Hyperparameter	**ECON**	**SKEW**	**GaussianMixture**	**Student-t**
`D`	100	100	250	150
`gamma`	0.0114	0.0114	0.00106	0.00106
`n_estimators_phi`	32	32	258	258
`n_estimators_psi`	913	913	517	517
`max_depth_phi`	5	5	3	3
`max_depth_psi`	3	3	1	1
`subsample_phi`	0.7	0.7	0.9	0.9
`subsample_psi`	0.7	0.7	0.7	0.7
`colsample_bytree_phi`	0.95	0.95	0.95	0.95
`colsample_bytree_psi`	0.95	0.95	0.95	0.95
`reg_alpha_phi`	0.00544	0.00544	0.0138	0.0138
`reg_alpha_psi`	0.00221	0.00221	0.00019	0.00019
`learning_rate_phi`	0.0928	0.0928	0.00624	0.00624
`learning_rate_psi`	0.0128	0.0128	0.0820	0.0820
`init_random`	orthonormal_sines	orthonormal_sines	orthonormal_sines	orthonormal_sines

Computational resources. All experiments were conducted on a machine running Ubuntu 22.04 LTS, equipped with an Intel Core i7-11800H CPU (2.3 GHz, 16 cores) and 16 GB of RAM. No GPU acceleration was used; all experiments were performed on CPU. The experimental pipeline was implemented in Python 3.12, using NumPy 1.26, scikit-learn 1.6, and XGBoost 3.1.3.