Causal Modeling with Stationary Diffusions

Lars Lorch
Department of Computer Science
ETH Zürich, Switzerland

Andreas Krause*
Department of Computer Science
ETH Zürich, Switzerland

Bernhard Schölkopf*
MPI for Intelligent Systems
Tübingen, Germany

Abstract

We develop a novel approach towards causal inference. Rather than structural equations over a causal graph, we learn stochastic differential equations (SDEs) whose stationary densities model a system's behavior under interventions. These stationary diffusion models do not require the formalism of causal graphs, let alone the common assumption of acyclicity. We show that in several cases, they generalize to unseen interventions on their variables, often better than classical approaches. Our inference method is based on a new theoretical result that expresses a stationarity condition on the diffusion's generator in a reproducing kernel Hilbert space. The resulting *kernel deviation from stationarity (KDS)* is an objective function of independent interest.[1]

1 Introduction

Decision-making, e.g., in the life and social sciences, requires predicting the outcomes of *interventions* in a system. Causal models characterize interventions as changes to the data-generating process, which enables us to reason about their downstream effects. By utilizing observed interventions, we can learn causal models that may generalize to predicting the effects of unseen interventions at test-time.

Classically, causal inference models a system $\mathbf{x} \in \mathbb{R}^d$ with a structural causal model (SCM) (Pearl, 2009)

$$\mathbf{x} = f(\mathbf{x}, \boldsymbol{\epsilon}) \,, \tag{1}$$

where $\epsilon_j \in \mathbb{R}$ are exogenous noise variables, and often under an additive noise assumption as $x_j = f_j(\mathbf{x}) + \epsilon_j$. Interventions can be realized as modifications of the functions f_j or the noise ϵ_j, and the SCM enables us to estimate the entailed distribution shifts in $\mathbf{x}$. However, since x_j depends recursively on $\mathbf{x}$, SCMs are generally limited to modeling *acyclic* causal effects.

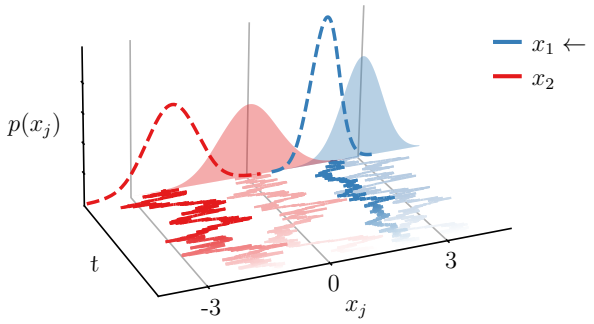


Figure 1: **Stationary SDEs as causal models.** The bottom axes show sample paths of a stationary diffusion in $\mathbb{R}^2$ before (pale) and after (dark) an intervention on the SDE governing x_1. The marginals $p(x_j)$ visualize the distribution shift in $p(x_1, x_2)$.

In this work, we propose to model a system's causal dependencies and their entailed probability distributions with stochastic differential equations (SDEs) and their entailed *stationary* densities. Specifically, we replace the SCM modeling $\mathbf{x}$ by its continuous-time analogue

$$\mathrm{d}\mathbf{x}_t = f(\mathbf{x}_t)\mathrm{d}t + \sigma(\mathbf{x}_t)\mathrm{d}\mathbb{W}_t$$

with the functions f, σ and the Wiener process $\{\mathbb{W}_t\}$ defined later. Stationary SDEs induce a time-invariant stationary density μ over $\mathbb{R}^d$, while they internally unroll the causal dependencies of the variables over time t, akin to real-world processes. We model the distribution of the variables $\mathbf{x}$ by this density μ, even though *not* observing the underlying process $\{\mathbf{x}_t\}$ over time. As in SCMs, interventions may be modeled as modifications to f and σ; the SDEs characterize how the stationary density of $\mathbf{x}$ changes by propagating the perturbations through its functional causal mechanisms (Figure 1). In the following, we will argue that modeling causation using stationary diffusions has several benefits:

Proceedings of the 27th International Conference on Artificial Intelligence and Statistics (AISTATS) 2024, Valencia, Spain. PMLR: Volume 238.

*Equal supervision

[1] Code: https://github.com/larslorch/stadion

Cyclic systems Causal effects in SDEs unfold over time, so feedback loops among the variables x_j become well-defined. Feedback is ubiquitous, e.g., in biological, environmental, and engineering systems (Hasty et al., 2002; Cox et al., 2000; Åström and Murray, 2008), but SCMs a priori only allow cycles under strong model restrictions (e.g., Mooij et al., 2011; Hyttinen et al., 2012). Our results suggest that stationary diffusions are more accurate than SCMs at predicting the effects of interventions in cyclic systems, while equally competitive in acyclic settings.

Graph-free Since acyclicity is not a constraint, stationary SDEs do not require the formalism of a causal graph. When learning SCMs, graphs serve as a tool for avoiding cycles in the causal dependencies, which classically restricts methods to discrete optimization (e.g., Chickering, 2003). While recent works introduce continuous formulations of the acyclicity constraint (e.g., Zheng et al., 2018), they still perform constrained optimization over the space of acyclic graphs.

Flexible distribution and intervention models Unlike inference of causal graphs and SCMs, which often exploits the statistical properties of particular functions, exogenous noise, or interventions (e.g., Geiger and Heckerman, 1994; Shimizu et al., 2006), our learning algorithm for stationary diffusions is general and thus agnostic to the system and intervention model.

Gradient-based inference without sampling We derive a novel kernelized objective that translates a stationarity condition on a diffusion's generator into reproducing kernel Hilbert spaces. Using this objective, stationary SDEs can be inferred consistently via gradient-based optimization, without sampling rollouts of the model or backpropagating gradients through time.

To describe our approach, we first devote Section 2 to technical background on kernels, SDEs, and their generators. This material is essential for Section 3, where we derive the *kernel deviation from stationarity (KDS)*. The KDS, which later forms our foundation for inference, serves as a model-agnostic objective for fitting stationary SDEs to an empirical target density. Building on these results, Section 4 introduces stationary diffusions as causal models and describes how to infer them from interventional data using the KDS. Sections 5 and 6 conclude with related work and experiments.

2 Background

We write $\mathbf{x} \in \mathbb{R}^d$ bold-faced for vectors, $x_j \in \mathbb{R}$ when indexing them, $\mathbf{x}_t \in \mathbb{R}^d$ for $\mathbf{x}$ at time t, and $\{\mathbf{x}_t\}$ for stochastic processes. The space C^p contains all p-times continuously differentiable functions $h : \mathbb{R}^d \to \mathbb{R}$, and C_c^p denotes the compactly supported functions in C^p.

Kernels and reproducing kernel Hilbert spaces Throughout this work, let $k(\mathbf{x}, \mathbf{x}') : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ denote a positive definite kernel function that is four times differentiable. Additionally, let $\mathcal{H}$ be the reproducing kernel Hilbert space (RKHS) of functions $\mathbb{R}^d \to \mathbb{R}$ associated with the kernel k and equipped with the norm $\langle \cdot, \cdot \rangle_{\mathcal{H}}$. The RKHS $\mathcal{H}$ satisfies that $k(\cdot, \mathbf{x}) \in \mathcal{H}$ for all $\mathbf{x} \in \mathbb{R}^d$, where $k(\cdot, \mathbf{x})$ denotes the function obtained when fixing the second argument of k at $\mathbf{x}$. Moreover, the RKHS $\mathcal{H}$ also satisfies the *reproducing property* that $h(\mathbf{x}) = \langle h, k(\cdot, \mathbf{x}) \rangle_{\mathcal{H}}$ for all $\mathbf{x} \in \mathbb{R}^d$ and $h \in \mathcal{H}$. In other words, evaluations of RKHS functions $h \in \mathcal{H}$ are inner products in $\mathcal{H}$ and parameterized by the "feature map" $k(\cdot, \mathbf{x})$. More background on kernels and RKHSs is given by Schölkopf and Smola (2002).

Stochastic differential equations SDEs are a stochastic analogue to differential equations. Rather than functions, their solutions are stochastic processes $\{\mathbf{x}_t\}$, $\mathbf{x}_t \in \mathbb{R}^d$ called diffusions, which are sequences of random vectors indexed by t. For our purposes, the Wiener process (or Brownian motion) $\{\mathbb{W}_t\}$, $\mathbb{W}_t \in \mathbb{R}^b$ can be viewed as driving noise with independent increments $\mathbb{W}_{t+s} - \mathbb{W}_t \sim \mathcal{N}(0, s\mathbf{I})$, where usually $b = d$. General SDEs $\mathrm{d}\mathbf{x}_t = f(\mathbf{x}_t)\mathrm{d}t + \sigma(\mathbf{x}_t)\mathrm{d}\mathbb{W}_t$ with some $\mathbf{x}_0 \sim p_0$ contain a drift $f : \mathbb{R}^d \to \mathbb{R}^d$ and a diffusion function $\sigma : \mathbb{R}^d \to \mathbb{R}^{d \times b}$. We make the common assumption that f and σ are Lipschitz continuous, which ensures that the SDEs have a unique strong solution given the initial vector $\mathbf{x}_0$ (Øksendal, 2003, Theorem 5.2.1). Formally, we consider integrals of $\{\mathbb{W}_t\}$ under the Itô convention (Øksendal, 2003, Chapter 3).

The diffusion $\{\mathbf{x}_t\}$ solving the SDEs is *stationary* if the probability density $\mu_t(\mathbf{x})$ of $\mathbf{x}_t$ at time t is the same for all $t \geq 0$ (Ethier and Kurtz, 1986, Chapter 4, Lemma 9.1). For example, the Ornstein-Uhlenbeck process solving $\mathrm{d}\mathbf{x}_t = -\mathbf{x}_t\mathrm{d}t + \sqrt{2}\mathrm{d}\mathbb{W}_t$ has the stationary density $\mu(\mathbf{x}) = \mathcal{N}(\mathbf{x}; \mathbf{0}, \mathbf{I})$. The so-called Fokker-Planck equation characterizes the time evolution of $\partial/\partial t\, \mu_t(\mathbf{x})$. Øksendal (2003) provides a more formal exposition of SDEs and the Brownian motion.

The infinitesimal generator The local evolution of a diffusion is described by its infinitesimal generator, which will play a central role in the derivation of our learning objective later on. The generator $\mathcal{A}$ associated to a stochastic process $\{\mathbf{x}_t\}$ is a linear *operator* that maps functions $h : \mathbb{R}^d \to \mathbb{R}$ to functions of the same signature. $\mathcal{A}$ can be viewed as the derivative of the semigroup of transition operators $\{\mathcal{T}_t : t \geq 0\}$ given by $(\mathcal{T}_t h)(\cdot) = \mathbb{E}_{\{\mathbf{x}_t\}}[h(\mathbf{x}_t) \,|\, \mathbf{x}_0 = \cdot\,]$ and is defined as

$$(\mathcal{A}h)(\mathbf{x}) := \lim_{t \downarrow 0} \frac{\mathcal{T}_t h(\mathbf{x}) - h(\mathbf{x})}{t} \tag{2}$$

for all functions $h \in \mathrm{dom}(\mathcal{A})$. The *domain* $\mathrm{dom}(\mathcal{A})$ of the generator contains all functions for which this limit exists for all $\mathbf{x} \in \mathbb{R}^d$ (Ethier and Kurtz, 1986, Chapter 1.1). The operators $\{\mathcal{T}_t : t \geq 0\}$ form a semigroup because $\mathcal{T}_0$ is the identity and $\mathcal{T}_s\mathcal{T}_t = \mathcal{T}_{s+t}$. Intuitively, the generator tells us how $h(\mathbf{x}_t)$ changes infinitesimally over time t when $\mathbf{x}_0 = \mathbf{x}$ (in expectation and given an arbitrary h). By Taylor's theorem, we can express $\mathcal{T}_t h$ as $\mathcal{T}_t h(\mathbf{z}) = h(\mathbf{z}) + t\,\mathcal{A}h(\mathbf{z}) + o(t)$ for small t.

If the stochastic process $\{\mathbf{x}_t\}$ solves the system of SDEs $\mathrm{d}\mathbf{x}_t = f(\mathbf{x}_t)\mathrm{d}t + \sigma(\mathbf{x}_t)\mathrm{d}\mathbb{W}_t$, then its generator $\mathcal{A}$ can be expressed in terms of f and σ for a large class of functions h. Specifically, for all functions $h \in C_c^2$, we have $\mathcal{A} = \mathcal{L}$ and $h \in \mathrm{dom}(\mathcal{A})$, where $\mathcal{L}$ is the linear differential operator $\mathcal{L}$ given by

$$(\mathcal{L}h)(\mathbf{x}) := f(\mathbf{x}) \cdot \nabla_{\mathbf{x}} h(\mathbf{x}) + \tfrac{1}{2} \operatorname{tr}\big(\sigma(\mathbf{x})\sigma(\mathbf{x})^\top \nabla_{\mathbf{x}} \nabla_{\mathbf{x}} h(\mathbf{x})\big) \tag{3}$$

(Øksendal, 2003, Theorem 7.3.3). For diagonal σ, the trace in (3) reduces to a weighted sum of the unmixed second-order partial derivatives of h, and for $\sigma = \mathbf{I}$, to the Laplacian $\Delta_{\mathbf{x}} h(\mathbf{x})$.

3 The Kernel Deviation from Stationarity

Suppose we are given a target density μ over $\mathbf{x} \in \mathbb{R}^d$. How can we learn the functions f and σ of a general system of SDEs $\mathrm{d}\mathbf{x}_t = f(\mathbf{x}_t)\mathrm{d}t + \sigma(\mathbf{x}_t)\mathrm{d}\mathbb{W}_t$ such that the diffusion solving the SDEs has the stationary density μ? In this first part, we will study this general inference question without yet considering causality and interventions in SDEs. Our starting point is a well-known link between the generator of a stochastic process and its stationary density. For a stochastic process $\{\mathbf{x}_t\}$, the density μ is the stationary density of $\{\mathbf{x}_t\}$ if and only if the generator $\mathcal{A}$ associated to $\{\mathbf{x}_t\}$ satisfies

$$\mathbb{E}_{\mathbf{x}\sim\mu}\big[\mathcal{A}h(\mathbf{x})\big] = 0 \tag{4}$$

for all functions h in a *core* for the generator $\mathcal{A}$ (Ethier and Kurtz, 1986, Chapter 4, Proposition 9.2). Roughly speaking, a core is a dense subset of functions in the domain $\mathrm{dom}(\mathcal{A})$ such that, if (4) holds for all functions h in the core, then (4) also holds for $h \in \mathrm{dom}(\mathcal{A})$ (see Hansen and Scheinkman, 1995, Section 6). Equation (4) states that every function h of $\{\mathbf{x}_t\}$ must have zero rate of change $\mathcal{A}h(\mathbf{x})$ in expectation over initializations by the stationary density μ. In other words, any $h(\mathbf{x}_t)$ must be, in expectation, invariant with time t iff μ is stationary.

If we can verify that the expected infinitesimal change over a target density μ is zero for an expressive class of test functions h (or conversely, learn a system of SDEs satisfying this), we may conclude that μ is a stationary density of the solution $\{\mathbf{x}_t\}$ to the SDEs. This insight suggests that it is sufficient to find the function w achieving the *largest* deviation from $\mathbb{E}_{\mathbf{x}\sim\mu}[\mathcal{A}h(\mathbf{x})] = 0$ among all test functions h. In the following, we derive a closed form for this maximum deviation over a sufficiently-rich, *infinite* set of functions as well as the witness function w achieving this maximum. We sketch our proofs and defer their formal arguments to Appendix B.

3.1 Bounding the Deviation from Stationarity

Our key idea for bounding the functional in (4) is to consider functions h in an RKHS $\mathcal{H}$. We show that this allows us to derive a closed-form expression for the supremum of (4) over an expressive, infinite subset of functions in the RKHS. In the following, let $\mathcal{H}$ be the RKHS of a kernel k as introduced in Section 2, and let $\mathcal{F} := \{h \in \mathcal{H} : \|h\|_{\mathcal{H}} \leq 1\}$ be the unit ball of $\mathcal{H}$.

To begin, we first focus on the closely-related functional $\mathbb{E}_{\mathbf{x}\sim\mu}[\mathcal{L}h(\mathbf{x})]$ involving the operator $\mathcal{L}$ instead of the generator $\mathcal{A}$ (Section 2). Recall that the operator $\mathcal{L}$ coincides with the generator $\mathcal{A}$ of the diffusion $\{\mathbf{x}_t\}$ solving $\mathrm{d}\mathbf{x}_t = f(\mathbf{x}_t)\mathrm{d}t + \sigma(\mathbf{x}_t)\mathrm{d}\mathbb{W}_t$ when applied to the well-behaved functions C_c^2. For this functional, we can show that there exists a representer function $g_{\mu,\mathcal{L}}$ in the RKHS $\mathcal{H}$, whose inner product with any function $h \in \mathcal{H}$ allows evaluating the functional:

Lemma 1 *Let μ be a probability density over $\mathbb{R}^d$ and assume that the functions f, σ, and the partial[2] derivatives $\partial/\partial x_{i,i} k(\mathbf{x},\mathbf{x})$ and $\partial^2/\partial x_{i,i}\partial x_{j,j} k(\mathbf{x},\mathbf{x})$ are square-integrable with respect to μ. Then, there exists a unique function $g_{\mu,\mathcal{L}} \in \mathcal{H}$ satisfying*

$$\mathbb{E}_{\mathbf{x}\sim\mu}[\mathcal{L}h(\mathbf{x})] = \langle h, g_{\mu,\mathcal{L}}\rangle_{\mathcal{H}}$$

for any $h \in \mathcal{H}$. Moreover, $g_{\mu,\mathcal{L}}(\cdot) = \mathbb{E}_{\mathbf{x}\sim\mu}[\mathcal{L}_{\mathbf{x}} k(\mathbf{x},\cdot)]$. Here, the notation $\mathcal{L}_{\mathbf{x}}$ indicates that $\mathcal{L}$ is applied to the argument $\mathbf{x}$.

To prove Lemma 1, we show that $\mathbb{E}_{\mathbf{x}\sim\mu}\mathcal{L}$ is a continuous linear functional and then invoke Riesz' representation theorem to prove that $g_{\mu,\mathcal{L}}$ exists. The reproducing property of $\mathcal{H}$ yields the explicit form of $g_{\mu,\mathcal{L}}$. Crucially, the representation in Lemma 1 allows us to derive a closed form for the supremum of $\mathbb{E}_{\mathbf{x}\sim\mu}[\mathcal{L}h(\mathbf{x})]$ over the unit ball $\mathcal{F}$, because the inner product with functions of $\mathcal{F}$ is maximized by the unit-norm function aligned with $g_{\mu,\mathcal{L}}$, that is, by $w_{\mu,\mathcal{L}} := g_{\mu,\mathcal{L}}/\|g_{\mu,\mathcal{L}}\|_{\mathcal{H}}$. Their inner product is then $\langle g_{\mu,\mathcal{L}}/\|g_{\mu,\mathcal{L}}\|_{\mathcal{H}}, g_{\mu,\mathcal{L}}\rangle_{\mathcal{H}} = \|g_{\mu,\mathcal{L}}\|_{\mathcal{H}}$. We will refer to the square of this RKHS norm as the *kernel deviation from stationarity* $\mathrm{KDS}(\mathcal{L},\mu;\mathcal{F})$:

[2] Like Steinwart and Christmann (2008), we use $\partial/\partial x_{i,i}$ to denote the first-order partial derivative w.r.t. both function arguments, so $\partial/\partial x_{i,i} k(\mathbf{x},\mathbf{x}) := \partial/\partial u_i \partial/\partial v_i k(\mathbf{u},\mathbf{v})|_{\mathbf{u}:=\mathbf{x},\mathbf{v}:=\mathbf{x}}$.

Theorem 2 *Under the assumptions of Lemma 1,*

$$\sup_{h\in\mathcal{F}} \mathbb{E}_{\mathbf{x}\sim\mu}\big[\mathcal{L}h(\mathbf{x})\big] = \sqrt{\mathrm{KDS}(\mathcal{L},\mu;\mathcal{F})}\,,$$

where $\mathrm{KDS}(\mathcal{L},\mu;\mathcal{F}) := \mathbb{E}_{\mathbf{x}\sim\mu}[\mathcal{L}_{\mathbf{x}}\,\mathbb{E}_{\mathbf{x}'\sim\mu}[\mathcal{L}_{\mathbf{x}'}k(\mathbf{x},\mathbf{x}')]]$. *Under additional regularity conditions on the functions* f,σ,k, *and* μ, *we may interchange the limits involved in the differentiation and integration operators and write* $\mathrm{KDS}(\mathcal{L},\mu;\mathcal{F}) = \mathbb{E}_{\mathbf{x}\sim\mu,\mathbf{x}'\sim\mu}\big[\mathcal{L}_{\mathbf{x}}\mathcal{L}_{\mathbf{x}'}k(\mathbf{x},\mathbf{x}')\big]$.

When thinking of $\mathcal{L}$ as the generator $\mathcal{A}$, the witness $w_{\mu,\mathcal{L}}$ is the smooth RKHS function that is subject to the largest infinitesimal change in the diffusion when initialized in expectation over μ; the functions in $\mathcal{F}$ are smooth, since their RKHS norm is limited to 1. Moreover, the KDS measures the maximal absolute deviation from (4) of any function in $\mathcal{F}$. While $\mathbb{E}_{\mathbf{x}\sim\mu}[\mathcal{L}h(\mathbf{x})]$ can be negative, $\sup_{h\in\mathcal{F}}\mathbb{E}_{\mathbf{x}\sim\mu}[\mathcal{L}h(\mathbf{x})]$ is always nonnegative, not only because it is equal to the RKHS norm $\|g_{\mu,\mathcal{L}}\|_{\mathcal{H}} \geq 0$. By the linearity of the functional $\mathcal{L}$, we also have $\mathbb{E}_{\mathbf{x}\sim\mu}[\mathcal{L}(-h)(\mathbf{x})] = -\mathbb{E}_{\mathbf{x}\sim\mu}[\mathcal{L}h(\mathbf{x})]$, and since $h\in\mathcal{F}$ iff $-h\in\mathcal{F}$, the supremum over $\mathcal{F}$ is nonnegative.

The KDS thus expresses the maximum discrepancy between $\mathcal{L}$ and a target μ in a kernelized, closed form over $\mathcal{F}$. We leverage this perspective later for learning stationary SDEs from data, since the SDE functions f and σ enter the KDS through the operator $\mathcal{L}$. Before doing so, we investigate whether the KDS of an RKHS $\mathcal{H}$ sufficiently discriminates among SDE models.

3.2 Consistency

While the KDS measures a deviation from stationarity, it may not be consistent—$\mathrm{KDS}(\mathcal{L},\mu;\mathcal{F}) = 0$ may not guarantee that all SDEs entailing the operator $\mathcal{L}$ indeed induce the stationary density μ. Guaranteeing this requires that the equality of the functional of $\mathcal{A}$ in (4) holds for all functions in a core for $\mathcal{A}$. However, the SDE-parameterized operator $\mathcal{L}$ only coincides with the generator $\mathcal{A}$ of the diffusion for all $h\in C_c^2$ (Section 2). Moreover, $\mathcal{F}$ may not be dense in a core for $\mathcal{A}$ and thus fail to be sufficiently rich for testing the condition in (4).

To link the KDS to $\mathcal{A}$, we need to relate a core for $\mathcal{A}$ to the functions spanned by the RKHS $\mathcal{H}$. In general, the relationship between these two function spaces strongly depends on the generality of the functions f,σ defining the SDEs (Ethier and Kurtz, 1986, Chapter 8) and the kernel k (Christmann and Steinwart, 2010; Kanagawa et al., 2018).[3] In the following, we show the consistency of the KDS for the Matérn kernel $k_{\nu,\gamma}$, which generalizes the Gaussian kernel $k_\gamma(\mathbf{x},\mathbf{x}') = \exp(-\|\mathbf{x}-\mathbf{x}'\|_2^2/2\gamma^2)$ (Appendix A). We achieve this by showing that a core for $\mathcal{A}$ is dense in the Matérn RKHS with respect to a Sobolev norm. Building on this, we then prove that, for any h in the core, there always exists a nearby RKHS element ensuring that $\mathbb{E}_{\mathbf{x}\sim\mu}[\mathcal{A}h(\mathbf{x})]$ is arbitrarily small:

Theorem 3 *Let* $k_{\nu,\gamma}$ *be a Matérn kernel with* $\nu > 2$ *defined over* $\mathbb{R}^d$, *and let* $\mathcal{F}$ *be the unit ball of its RKHS. Let* μ *be a probability density over* $\mathbb{R}^d$ *and* f,σ *be bounded functions with* $\sigma\sigma^\top$ *positive definite that define the SDEs* $\mathrm{d}\mathbf{x}_t = f(\mathbf{x}_t)\mathrm{d}t + \sigma(\mathbf{x}_t)\mathrm{d}\mathbb{W}_t$. *Then,* μ *is a stationary density of the stochastic process* $\{\mathbf{x}_t\}$ *solving the SDEs if and only if*

$$\mathrm{KDS}(\mathcal{L},\mu;\mathcal{F}) = 0\,.$$

Our result shows that the KDS is zero if and only if a system of SDEs with bounded functions f and σ induces the stationary density μ. The boundedness assumption allows us to leverage established results in stochastic analysis on cores of generators of diffusions. However, more general results on cores may exist and imply the consistency of the KDS for other classes of kernels or SDEs. In applications, we usually work with continuous functions f and σ, which can only be unbounded as $\|\mathbf{x}\|\to\infty$. In practice, this occurs with arbitrarily low probability if a system is stationary, so we may think of Theorem 3 as informally extending to the case of continuous functions, even though not formally covered by the assumptions.

It is worth noting that SDEs may not have *unique* stationary densities in general. Different initial distributions of the random vector $\mathbf{x}_0$ may result in different stationary densities of the solution $\{\mathbf{x}_t\}$. The Lipschitz assumptions in Section 2 only guarantee that $\{\mathbf{x}_t\}$ is the unique strong solution given $\mathbf{x}_0$. From our inference viewpoint, however, we always initialize $\mathbf{x}_0\sim\mu$ with the target μ, which ensures that our SDE models have unique stationary densities. Under additional assumptions, stationary densities can be shown to be unique for any $\mathbf{x}_0$ (see Khasminskii, 2011, Section 4.4).

3.3 The KDS as a Learning Objective

The KDS provides a closed-form expression for the maximum stationarity violation of any $h\in\mathcal{F}$. Since it quantifies this violation (as an RKHS norm), the KDS serves as an objective we can minimize to fit a system of SDEs to a target density μ. Specifically, given a dataset $D = \{\mathbf{x}^{(1)},\ldots,\mathbf{x}^{(N)}\}$ of i.i.d. samples $\mathbf{x}^{(n)}\sim\mu$, we can compute an unbiased empirical estimate of the $\mathrm{KDS}(\mathcal{L},\mu;\mathcal{F})$ with the U-statistic given by

$$\hat{\mathrm{KDS}}(\mathcal{L},D;k) := \tfrac{1}{N(N-1)}\sum_{m=1}^{N}\sum_{n\neq m}^{N}\mathcal{L}_{\mathbf{x}}\mathcal{L}_{\mathbf{x}'}k(\mathbf{x}^{(m)},\mathbf{x}^{(n)}) \tag{5}$$

[3] Universal kernels (Micchelli et al., 2006) are of limited use here, since the denseness of universal RKHSs is usually established in supremum norm, not for the partial derivatives in $\mathcal{L}$, and diffusions are defined over $\mathbb{R}^d$ (noncompact).

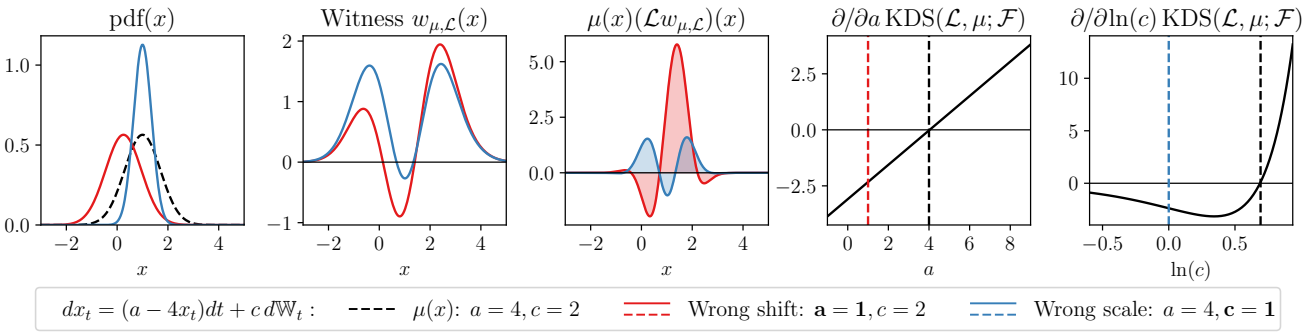


Figure 2: **Components of the KDS for a stationary linear SDE** and a Gaussian kernel k_γ with $\gamma = 0.5$. Expectations over μ are approximated with 1000 samples. *1:* Densities of a target (μ, black) and two alternative models. *2:* KDS witness functions for the misspecified models. *3:* Witnesses after applying $\mathcal{L}$, yielding their time derivatives in the diffusion. After multiplying by μ, the KDS is equal to the integral of the shaded areas. *4-5:* KDS derivatives with respect to a and c, fixing the other parameters at those of the target model. The partial derivatives have zeroes at the true parameters of the model inducing μ, thus gradient descent drives the incorrect a and c to their true values (indicated by vertical, dashed lines).

In Appendix C.1, we also provide an unbiased estimate of the KDS that scales *linearly* in N, which can be useful in large-scale applications.

When the SDE model $f_{\boldsymbol{\theta}}, \sigma_{\boldsymbol{\theta}}$ is parameterized by $\boldsymbol{\theta}$, we will indicate the dependency of the operator $\mathcal{L}$ by a superscript (here as $\mathcal{L}^{\boldsymbol{\theta}}$). The KDS depends on the SDE parameters $\boldsymbol{\theta}$ through the operator $\mathcal{L}^{\boldsymbol{\theta}}$ since $\mathcal{L}^{\boldsymbol{\theta}} h(\mathbf{x}) = f_{\boldsymbol{\theta}}(\mathbf{x}) \cdot \nabla_{\mathbf{x}} h(\mathbf{x}) + \frac{1}{2}\mathrm{tr}(\sigma_{\boldsymbol{\theta}}(\mathbf{x})\sigma_{\boldsymbol{\theta}}(\mathbf{x})^\top \nabla_{\mathbf{x}} \nabla_{\mathbf{x}} h(\mathbf{x}))$. Minimizing the KDS thus enables us to estimate the parameters of a stochastic dynamical system without backpropagating gradients through time. Moreover, the function $\mathcal{L}^{\boldsymbol{\theta}}_{\mathbf{x}} \mathcal{L}^{\boldsymbol{\theta}}_{\mathbf{x}'} k(\mathbf{x}, \mathbf{x}')$ inside the KDS is fully differentiable with respect to $\boldsymbol{\theta}$. Notably, the KDS is exact up to the sample approximation of the expectations over the target μ in (5)—there are no SDE model components we need to sample from, roll out, reparameterize, or approximate.

It is instructive to consider the special case of $\sigma = \mathbf{I}$. The function $\mathcal{L}^{\boldsymbol{\theta}}_{\mathbf{x}} \mathcal{L}^{\boldsymbol{\theta}}_{\mathbf{x}'} k$ inside the KDS is then given by

$$\begin{aligned} \mathcal{L}^{\boldsymbol{\theta}}_{\mathbf{x}} \mathcal{L}^{\boldsymbol{\theta}}_{\mathbf{x}'} k(\mathbf{x}, \mathbf{x}') &= f_{\boldsymbol{\theta}}(\mathbf{x}) \cdot \nabla_{\mathbf{x}} \nabla_{\mathbf{x}'} k(\mathbf{x}, \mathbf{x}') \cdot f_{\boldsymbol{\theta}}(\mathbf{x}') \\ &\quad + \tfrac{1}{2} f_{\boldsymbol{\theta}}(\mathbf{x}) \cdot \nabla_{\mathbf{x}} \Delta_{\mathbf{x}'} k(\mathbf{x}, \mathbf{x}') \\ &\quad + \tfrac{1}{2} f_{\boldsymbol{\theta}}(\mathbf{x}') \cdot \nabla_{\mathbf{x}'} \Delta_{\mathbf{x}} k(\mathbf{x}, \mathbf{x}') \\ &\quad + \tfrac{1}{4} \Delta_{\mathbf{x}} \Delta_{\mathbf{x}'} k(\mathbf{x}, \mathbf{x}') \,, \end{aligned} \tag{6}$$

where $\Delta_{\mathbf{x}} := \mathrm{tr}\, \nabla_{\mathbf{x}} \nabla_{\mathbf{x}}$ is the Laplacian. This expression contains a matrix, two vectors, and a scalar involving k that are all independent of $\boldsymbol{\theta}$. The kernel terms may thus be reused, e.g., during gradient descent on $\boldsymbol{\theta}$. For general $\sigma_{\boldsymbol{\theta}}$, there also exists an explicit expression, but it may be easier to leverage the operator view of $\mathcal{L}^{\boldsymbol{\theta}}$ to compute $\mathcal{L}^{\boldsymbol{\theta}}_{\mathbf{x}} \mathcal{L}^{\boldsymbol{\theta}}_{\mathbf{x}'} k$ and its gradients with automatic differentiation. We provide the explicit form and pseudocode demonstrating this case in Appendix C.2.

Example Figure 2 illustrates how the KDS may be used to learn the SDE parameters $\boldsymbol{\theta}$. We consider a target model $dx_t = (a + bx_t)dt + c\mathrm{d}\mathbb{W}_t$ with the closed-form density $\mu(x) = \mathcal{N}(x; -a/b, -c^2/2b)$ for $b < 0$ and $c > 0$ (Jacobsen, 1993). We use the KDS, approximated by samples from μ, to measure the fit of two models with incorrect a and c controlling the mean and variance, respectively. The partial derivatives of the KDS have zeroes at the true parameters of the model inducing μ and can thus be inferred with gradient descent.

4 Stationary Diffusions as Causal Models

In this section, we describe how stationary diffusions can serve as causal models. To facilitate this exposition, we first leave the KDS aside and focus on discussing causality in SDEs, intervention models, and related properties. To conclude, we then leverage the KDS as an objective for learning stationary diffusions as causal models from a collection of interventional datasets.

4.1 Modeling Causal Dependencies with Stationary SDEs

Probabilistic causal models of a system $\mathbf{x} \in \mathbb{R}^d$ entail more than the *observational* density of the variables. A causal model contains additional information that characterizes the *interventional* densities of the system under interventions on its data-generating process (Peters et al., 2017). This information may be in the form of, say, functions f_j that explicitly model the densities of x_j and remain invariant under interventions elsewhere, as in SCMs. Which causal model of a system and which intervention model are adequate depends on the

application and the level of modeling granularity (Hoel et al., 2013; Rubenstein et al., 2017; Schölkopf, 2022).

In this work, we propose to characterize the causal dependencies among the variables $\mathbf{x}$ by modeling a stationary dynamical system $\mathbf{x}_t \in \mathbb{R}^d$ underlying the generative process of $\mathbf{x}$. Specifically, we model the distribution of the variables $\mathbf{x}$ with the stationary density μ of a process $\mathbf{x}_t$, which evolves according to the SDEs

$$\mathrm{d}\mathbf{x}_t = f_{\boldsymbol{\theta}}(\mathbf{x}_t)\mathrm{d}t + \sigma_{\boldsymbol{\theta}}(\mathbf{x}_t)\mathrm{d}\mathbb{W}_t\,, \tag{7}$$

where $\boldsymbol{\theta} \in \mathbb{R}^k$ are parameters and μ is the observational stationary density, i.e., $\mathbf{x}_t \sim \mu$. The core idea is that introducing an explicit time dimension enables propagating feedback cycles in the causal dependencies of the variables—even though we do *not* observe the system $\mathbf{x}$ itself as a time series. Through stationarity, time remains remains internal to the SDE model. In contrast to stationary SDEs, SCMs do not allow for cycles in the causal structure except under restrictive model and invertibility assumptions (see *Related Work* in Section 5).

Similar to the structural equations in SCMs, the differential equations in SDEs provide a *mechanistic* (or functional) model of the causal dependencies among the variables $\mathbf{x}$ (Peters et al., 2017; Schölkopf, 2022). In other words, f_j and σ_j model which variables in $\mathbf{x}$ affect the variable x_j via an explicit functional dependency that holds independent of perturbations of the variables or the functions governing the other variables. However, both SCMs and stationary SDEs should be thought of as phenomenological abstractions of the true physical processes underlying the observables $\mathbf{x}$ (e.g., Peters et al., 2017, Section 2.3.3), with stationary SDEs characterizing the processes explicitly over time.

Following SCMs, we may graphically summarize the causal dependencies in stationary SDEs by reading off its direct functional dependencies, e.g., to gain qualitative insights or incorporate prior knowledge. However, neither modeling nor inference with stationary SDEs requires such a graph. Moreover, the graphs also imply different properties than for SCMs, e.g., the modeled distribution μ is not necessarily Markovian with respect to the graph (cf. Peters et al., 2017, Proposition 6.31).

4.2 Interventions

Interventions in SDEs can be modeled in various ways and often in analogy to SCMs (Eberhardt and Scheines, 2007). We formalize an intervention as transforming $f_{\boldsymbol{\theta}}$ and $\sigma_{\boldsymbol{\theta}}$ into the modified mechanisms $f_{\boldsymbol{\theta},\boldsymbol{\phi}}$ and $\sigma_{\boldsymbol{\theta},\boldsymbol{\phi}}$ with additional parameters $\boldsymbol{\phi}$ such that (7) evolves as $\mathrm{d}\mathbf{x}_t = f_{\boldsymbol{\theta},\boldsymbol{\phi}}(\mathbf{x}_t)\mathrm{d}t + \sigma_{\boldsymbol{\theta},\boldsymbol{\phi}}(\mathbf{x}_t)\mathrm{d}\mathbb{W}_t$ and induces the stationary density $\mu_{\boldsymbol{\phi}}$. For example, some real-world perturbations may be modeled as shift-scale interventions, in which $f_{\boldsymbol{\theta}}(\cdot)_j$ and $\sigma_{\boldsymbol{\theta}}(\cdot)_j$ of a variable x_j are shifted and scaled by δ, β, respectively, as

$$f_{\boldsymbol{\theta},\boldsymbol{\phi}}(\mathbf{x})_j = f_{\boldsymbol{\theta}}(\mathbf{x})_j + \delta \;\text{ and }\; \sigma_{\boldsymbol{\theta},\boldsymbol{\phi}}(\mathbf{x})_j = \beta\,\sigma_{\boldsymbol{\theta}}(\mathbf{x})_j\,, \tag{8}$$

where $\boldsymbol{\phi} = \{\delta, \beta\}$. Analogous shift interventions have been studied in acyclic and cyclic SCMs (Zhang et al., 2021; Rothenhäusler et al., 2015). Most settings assume that interventions are sparse, have known target variables, or few parameters (Schölkopf, 2022).

4.3 Properties

Complexity Stationary diffusions can be modeled by general functions f and σ. Even with diagonal $\sigma = \sqrt{2}\mathbf{I}$, they can characterize any observational density μ via its score function $f = -\nabla_{\mathbf{x}} \log \mu$ (as a Langevin diffusion). When σ is non-diagonal, the driving noise of the equations $\mathrm{d}\mathbf{x}_t$ becomes correlated, which can model confounding. Thus, the function classes of f and σ determine the complexity of the densities modeled by the diffusion, not the Brownian motion $\{\mathbb{W}_t\}$. Besides some notable exceptions (Immer et al., 2023), the assumptions of stationary diffusions are less restrictive than those of SCMs, where the exogenous noise defines the distributional family *a priori*.

Stability Using diffusions for causal modeling relies on the stationarity, i.e., stability, of the SDEs. For general $f_{\boldsymbol{\theta}}$ and $\sigma_{\boldsymbol{\theta}}$, stability is not guaranteed, particularly when randomly initializing the model parameters $\boldsymbol{\theta}$. For example, in linear systems $\mathrm{d}\mathbf{x}_t = (\mathbf{a} + \mathbf{B}\mathbf{x}_t)\mathrm{d}t + \mathbf{C}\mathrm{d}\mathbb{W}_t$, stability requires that the eigenvalues of $\mathbf{B}$ have negative real parts (Särkkä and Solin, 2019). Guaranteeing stability under interventions, however, is possible in certain cases: in linear systems, the shift-scale interventions in (8) do not affect stability. More generally, Theorem 3 shows that $\mathrm{KDS} = 0$ can guarantee stability and act as a certificate, even for complex model classes.

Identifiability Causal modeling aims at generalizing to (combinations of) intervention classes when learning a model from a set of observed interventions. Generalizing to unseen perturbations may not require fully identifying $\boldsymbol{\theta}$. For SDEs in particular, a density μ does not uniquely identify the true parameters $\boldsymbol{\theta}$ in a model class without unverifiable assumptions: changing the *speed* of a diffusion via $\mathrm{d}\mathbf{x}_t = sf(\mathbf{x}_t)\mathrm{d}t + \sqrt{s}\,\sigma(\mathbf{x}_t)\mathrm{d}\mathbb{W}_t$ for $s > 0$ leaves the stationary density unchanged. The operator $s\mathcal{L}$ satisfies the same stationarity conditions as $\mathcal{L}$ (Hansen and Scheinkman, 1995). While linear SDEs are identifiable up to speed scaling under specific sparsity conditions (Dettling et al., 2022), it is, to our knowledge, not yet known to what degree multiple interventional densities $\mu_{\boldsymbol{\phi}}$ identify stationary SDEs. As we investigate in Section 6, stationary diffusions empirically generalize to unseen interventions, hence weaker notions than parametric identifiability may be appropriate.

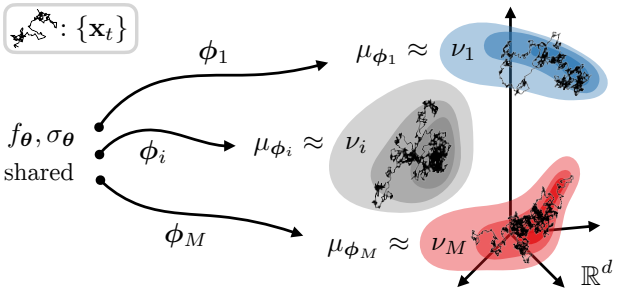


Figure 3: **Inference intuition.** Our goal is to infer mechanisms $f_{\boldsymbol{\theta}}, \sigma_{\boldsymbol{\theta}}$ that explain the observed densities $\nu_{1:M}$. To achieve this, we jointly learn $\boldsymbol{\theta}$ and interventions $\boldsymbol{\phi}_i$ that induce stationary densities $\mu_{\boldsymbol{\phi}_i}$ fitting ν_i.

4.4 Learning Causal Stationary Diffusions from Interventional Data

Suppose we observe M interventional densities ν_i of an unknown system $\mathbf{x}$. Given corresponding datasets $D_i \sim \nu_i$ of i.i.d. samples, our goal is to learn a causal model that allows predicting the effects of unseen interventions on the system. To achieve this, we seek to learn an SDE model $f_{\boldsymbol{\theta}}, \sigma_{\boldsymbol{\theta}}$, whose stationary densities fit the observed densities ν_i under a considered class of interventions $\boldsymbol{\phi}_i$ (Figure 3). If the model explains all densities ν_i using one set of causal mechanisms $f_{\boldsymbol{\theta}}, \sigma_{\boldsymbol{\theta}}$, adding sparsity and parameter regularization, we may expect it to generalize to unseen interventions.

Using the KDS, the model $\boldsymbol{\theta}$ and the intervention parameters $\boldsymbol{\phi}_{1:M}$ can be learned jointly from $\nu_{1:M}$ with gradient descent. We iteratively draw batches of the datasets D_i and then update $(\boldsymbol{\theta}, \boldsymbol{\phi}_i)$ using the KDS gradients of the intervened SDEs $f_{\boldsymbol{\theta},\boldsymbol{\phi}_i}$ and $\sigma_{\boldsymbol{\theta},\boldsymbol{\phi}_i}$. Jointly learning the interventions $\boldsymbol{\phi}_i$ alongside the model $\boldsymbol{\theta}$ is well-posed when we observe multiple ν_i and the interventions have few degrees of freedom (e.g., sparse targets, few parameters), since $\boldsymbol{\theta}$ is shared across all $\boldsymbol{\phi}_i$. To mitigate overfitting, we apply a *group lasso* penalty $R(\boldsymbol{\theta}_j)$ separately to the parameters $\boldsymbol{\theta}_j$ of each x_j, with appropriate groups depending on the model class, to encourage sparse causal dependencies (Yuan and Lin, 2006). Algorithm 1 summarizes the inference method.

When learning both $f_{\boldsymbol{\theta}}$ and $\sigma_{\boldsymbol{\theta}}$, the invariance to speed scaling described in Section 4.3 can cause an instability close to convergence, as decreasing the speed s via $sf_{\boldsymbol{\theta}}$ and $\sqrt{s}\sigma_{\boldsymbol{\theta}}$ shrinks the KDS. This can be prevented by fixing (the scale of) subsets of the parameters of $f_{\boldsymbol{\theta}}$ or $\sigma_{\boldsymbol{\theta}}$, for example, the self-regulating dependence of $f_{\boldsymbol{\theta}}(\mathbf{x})_j$ on x_j. Empirically, minimizing the KDS was sufficient in combination with sparsity regularization to ensure that the learned SDEs are stable upon convergence. However, future work could aim at guaranteeing stability directly through properties of the model class (Richards et al., 2018; Kolter and Manek, 2019).

Algorithm 1 Learning causal stat. diffs. via the KDS

Input: Interventional datasets $\{D_1, \ldots, D_M\}$, kernel k, sparsity penalty λ, optimizer

Initialize model $\boldsymbol{\theta}$ and interventions $\{\boldsymbol{\phi}_1, \ldots, \boldsymbol{\phi}_M\}$

while not converged **do**

Draw environment index $i \sim \mathrm{Unif}(\{1, \ldots, M\})$

Sample batch of interventional data $D \sim D_i$

Update $\boldsymbol{\theta}$ and $\boldsymbol{\phi}_i$ with optimizer step

$$\propto -\nabla_{\boldsymbol{\theta},\boldsymbol{\phi}_i}\Big(\hat{\mathrm{KDS}}(\mathcal{L}^{\boldsymbol{\theta},\boldsymbol{\phi}_i}, D; k) + \lambda \sum_{j=1}^{d} R(\boldsymbol{\theta}_j)\Big)$$

return $\boldsymbol{\theta}$ and $\{\boldsymbol{\phi}_1, \ldots, \boldsymbol{\phi}_M\}$

5 Related Work

Causality in dynamical systems When observing processes over time, fields like Granger causality (Granger, 1969), autoregressive modeling (Hyvärinen et al., 2010), and system identification (Ljung, 1998) allow inferring notions of causation. Recent works leverage continuous optimization for this (Pamfil et al., 2020; Tank et al., 2021) or study structure identification in differential equations (Bellot et al., 2022). Hansen and Sokol (2014) and Peters et al. (2022) formally study interventions in SDE systems observed over time. Contrary to these time series settings, we do *not* assume observations of a process over time. Instead, we adopt the novel perspective of using stationary distributions to model and infer causality. Our approach makes explicit that causal models, including SCMs, are abstractions of processes taking place in time (e.g., Peters et al., 2017, Section 2.3.3)—even when causation occurs on scales that either are not or cannot be measured as time series. Varando and Hansen (2020) also study stationary SDEs in the linear case, but they interpret them as probabilistic graphical models, not considering causality or interventions. Orthogonal to our work, Mooij et al. (2013), Blom et al. (2020), and Bongers et al. (2022) theoretically investigate how equilibria of differential equations relate to classical SCMs.

Cyclic graphical modeling Several works interpret SCMs in ways that allow cycles (Richardson, 1996; Lacerda et al., 2008; Mooij et al., 2011; Hyttinen et al., 2012; Mooij and Heskes, 2013; Rothenhäusler et al., 2015; Sethuraman et al., 2023). These approaches usually assume additive noise and linearity, sometimes with restrictions on the feedback, and require a unique solution $\mathbf{x}$ to $\mathbf{x} = f(\mathbf{x}) + \boldsymbol{\epsilon}$ given any possible $\boldsymbol{\epsilon}$ (Bongers et al., 2021). Our proposal of modeling causality with stationary SDEs shares the intuition of an equilibrium but expands on the insight that cyclicity necessarily introduces a notion of time. Departing from graphical models ultimately enables us to drop prior model re-

strictions and consider more general mechanisms and interventions. As real-world processes evolve in time, some challenge the notion of aggregating causality in graphical models altogether (Dawid, 2010; Aalen et al., 2016). In particular, causal mechanisms that are independent in temporal processes may not be translatable into static conditional independencies, and for that matter, graphical models (Tejada-Lapuerta et al., 2023).

Statistical inference and kernels The idea of producing diffusions that imply certain densities goes back to Wong (1964), who linked diffusions with polynomial SDE functions f and σ to the Pearson distributions. In econometrics, the infinitesimal generator and Equation (4) are known tools for fitting diffusion models, but usually with specific parameterizations and test functions (Hansen and Scheinkman, 1995; Conley et al., 1997; Duffie and Glynn, 2004; see Aït-Sahalia et al., 2010, Section 3, for an overview). The KDS extends these works by introducing a general-purpose characterization of stationarity that covers an infinite class of test functions in closed form. Our techniques establish novel connections between SDEs and RKHSs and build on kernel properties previously used by, for example, kernel mean embeddings (Smola et al., 2007), the MMD (Gretton et al., 2012), and the kernelized Stein discrepancy (Liu et al., 2016).

From a statistical perspective, the KDS can be viewed as a Stein discrepancy (Stein, 1972). These discrepancies measure the fit of a distribution ν to a target μ by constructing operators $\mathcal{S}_\mu$ that produce zero mean functions $\mathbb{E}_{\mathbf{x}\sim\nu}[\mathcal{S}_\mu h(\mathbf{x})]$ when $\nu = \mu$. One way of obtaining such an operator $\mathcal{S}_\mu$ is through the generator of a stationary Markov process (Barbour, 1988). The Langevin diffusion $\mathrm{d}\mathbf{x}_t = \nabla_{\mathbf{x}} \log \mu(\mathbf{x})\mathrm{d}t + \sqrt{2}\mathrm{d}\mathbb{W}_t$ has the stationary distribution μ, so one can use (4) to show that the Stein operator $(\mathcal{S}_\mu h)(\mathbf{x}) := \nabla_{\mathbf{x}} \log \mu(\mathbf{x}) h(\mathbf{x}) + \nabla_{\mathbf{x}} h(\mathbf{x})$ satisfies $\mathbb{E}_{\mathbf{x}\sim\mu}[\mathcal{S}_\mu h(\mathbf{x})] = \mathbf{0}$ for any h (Gorham and Mackey, 2015). By contrast, the KDS assigns opposite roles to the operator (now *model*, per $\mathcal{L}^{\boldsymbol{\theta}}$) and the distribution (now *target*). As a result, the KDS enables us to learn general SDEs $f_{\boldsymbol{\theta}}, \sigma_{\boldsymbol{\theta}}$ inducing a known distribution μ, instead of learning a distribution ν compatible with a known operator $\mathcal{S}_\mu$. This conversely implies that the KDS performs score estimation in the special case of the Langevin diffusion (Hyvärinen, 2005).

6 Experiments

6.1 Setup

The downstream purpose of causal modeling is to predict the effects of interventions in a system. To evaluate this, we compare the interventional densities predicted by stationary diffusions to those by existing methods. All methods first learn a causal model from interventional data with known targets and then predict the distributions resulting from unseen interventions by sampling data from the learned models. The test interventions are out-of-distribution, that is, performed on variables not intervened upon in the training data.

In the following, we summarize the experimental setup. Appendix D provides additional supplementary details on the data, methods, and metrics for benchmarking.

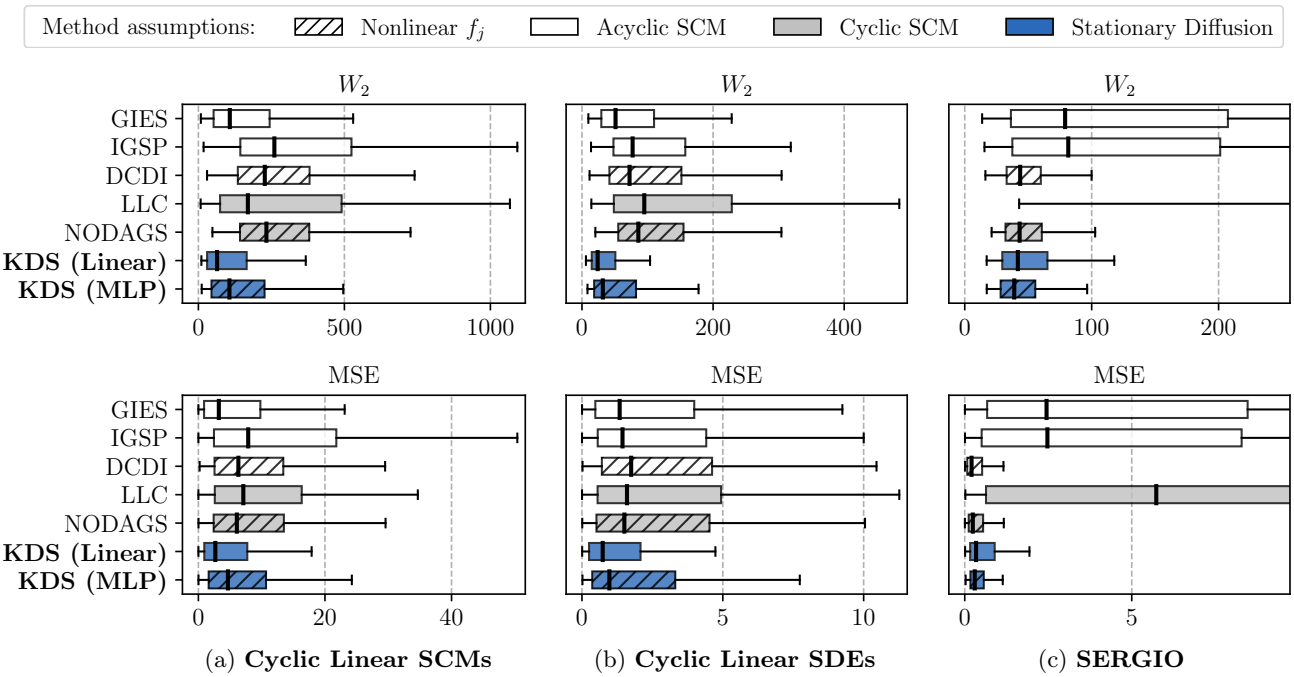
Data We evaluate the methods on data of sparse cyclic linear systems (SCMs and stationary SDEs) and expression data of sparse gene regulatory networks. For the latter, we simulate the SERGIO model by Dibaeinia and Sinha (2020), which requires acyclic dependencies, without technical noise. For each randomly-generated system, we sample observational data and interventional data for 10 train and 10 test interventions on single, disjoint target variables, each dataset containing 1000 observations. In the linear systems, we perform shift interventions; in SERGIO, we implement gain-of-function (overexpression) gene perturbations (e.g., Norman et al., 2019). All datasets are standardized by the mean and variance of the observational data.

Methods Using the KDS (Algorithm 1), we learn stationary diffusions with linear and multi-layer perceptron (MLP) mechanisms $f_{\boldsymbol{\theta}}(\mathbf{x})$, whose components are defined independently for each x_j as

$$f_{\boldsymbol{\theta}_j}(\mathbf{x})_j = b^j + \mathbf{w}^j \cdot \mathbf{x} \qquad \text{(Linear)}$$
$$f_{\boldsymbol{\theta}_j}(\mathbf{x})_j = b^j + \mathbf{w}^j \cdot g(\mathbf{U}^j\mathbf{x} + \mathbf{v}^j) - x_j \qquad \text{(MLP)}$$

where g is the sigmoid nonlinearity. The diffusion matrix is parameterized as $\sigma_{\boldsymbol{\theta}}(\mathbf{x}) = \mathrm{diag}(\exp(\boldsymbol{\sigma}))$. The interventions $\boldsymbol{\phi}$ that are jointly learned with $\boldsymbol{\theta}$ to fit the interventional data are modeled as shifts on the known target variables, as in (8, left). Appendix D defines the regularizers $R(\boldsymbol{\theta}_j)$ of both models. We use the Gaussian kernel k_γ to estimate the KDS. Test-time predictions are sampled from the SDEs via the Euler-Maruyama scheme (Appendix A.1).

We compare the stationary SDE models to five SCM approaches that use interventional data: score-based GIES (Hauser and Bühlmann, 2012), the constraint-based IGSP algorithm (Wang et al., 2017), both based on linear-Gaussian SCMs, and nonlinear DCDI (Brouillard et al., 2020). We also benchmark LLC (Hyttinen et al., 2012) and NODAGS (Sethuraman et al., 2023), which learn cyclic linear and nonlinear SCMs, respectively. The graph discovery methods use a maximum-likelihood linear SCM as the causal model. The hyperparameters of each method are tuned on validation splits of the interventional datasets later used for learning the evaluated models.



(a) **Cyclic Linear SCMs** (b) **Cyclic Linear SDEs** (c) **SERGIO**

Figure 4: **Benchmarking results** ($d = 20$ variables, Erdős-Rényi causal structure). Metrics are computed from 10 test interventions on unseen target variables in 50 randomly-generated systems. Box plots show medians and interquartile ranges (IQR). Whiskers extend to the largest value inside 1.5 times the IQR length from the boxes. Overall, causal stationary diffusions learned via the KDS (Algorithm 1, bold-faced) are the most accurate at predicting the effects of interventions on unseen targets, measured in terms of both W_2 (↓) and MSE (↓).

Metrics The test interventions performed to query the learned causal models are shift interventions that match the interventional mean of the target variable in the held-out data. To allow comparing methods with explicit and implicit densities, we report the Wasserstein distance W_2 between the true interventional data and the data sampled by the learned models under the queried interventions. We also report the mean squared error (MSE) of the true and predicted empirical means (Zhang et al., 2022).

6.2 Results

Figures 4a and 4b present the results for the cyclic linear SCM and stationary SDE systems, respectively. Both the linear and MLP diffusions learned via the KDS (Algorithm 1) achieve the most accurate interventional density predictions in both the W_2 and MSE metrics. The acyclic approaches, in particular GIES, show competitive performance, highlighting a trade-off between model complexity and the entailed inference challenge, even when the data qualitatively violates acyclicity. In contrast, the cyclic SCM approaches underperform, particularly LLC, whose model assumptions—apart from standardization of the data—perfectly align with this setting. The synthetic gene expression data assesses all methods under model mismatch. Figure 4c shows that stationary diffusions, especially the nonlinear MLP diffusion, match the best baselines DCDI and NODAGS, which also model nonlinearity. This highlights the potential of using stationary SDEs for causal modeling in complex data-generating processes, even in acyclic settings. Appendix E presents additional results for scale-free causal structures, which show similar findings overall, as well as significance tests. Compared to Figure 4, the MLP diffusion achieves worse MSE in linear systems but remains on par with the baselines.

7 Conclusion

We propose a new approach for modeling causality and interventional distributions using stationary SDEs. Similar to real-world processes, stationary diffusions unroll causal dependencies and feedback over time, yet the densities they model remain time-invariant. To propose a practical algorithm, we derive a tractable kernelized objective for learning stationary SDEs from data. We believe our results linking diffusions to RKHSs provide new fundamental tools for analyzing and learning diffusions, also beyond the context of causal modeling. Future directions include formally studying generalization under intervention classes, showing consistency of the KDS for more general models, and learning latent representations governed by stationary diffusions.

Acknowledgments

Many thanks to Ya-Ping Hsieh and Mohammad Reza Karimi for the engaging discussions on SDEs in the early stages of this work. We additionally thank Charlotte Bunne, Paweł Czyż, Jonas Rothfuss, and Scott Sussex for their helpful comments on versions of the manuscript. This work has also greatly benefited from discussions with Nicolas Emmenegger, Parnian Kassraie, Jonas Hübotter, Mojmír Mutný, Jonas Rothfuss, Zebang Shen, and Ingo Steinwart, for which we are very thankful.

This research was supported by the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation program grant agreement no. 815943 and the Swiss National Science Foundation under NCCR Automation, grant agreement 51NF40 180545.

References

Aalen, O. O., Røysland, K., Gran, J. M., Kouyos, R., and Lange, T. (2016). Can we believe the DAGs? A comment on the relationship between causal DAGs and mechanisms. *Statistical methods in medical research*, 25(5):2294–2314.

Adams, R. A. and Fournier, J. J. (2003). *Sobolev spaces*. Elsevier, 2nd edition.

Aït-Sahalia, Y., Hansen, L. P., and Scheinkman, J. A. (2010). Operator methods for continuous-time Markov processes. *Handbook of financial econometrics: tools and techniques*, pages 1–66.

Åström, K. J. and Murray, R. M. (2008). *Feedback systems: an introduction for scientists and engineers*. Princeton university press.

Barbour, A. D. (1988). Stein's method and Poisson process convergence. *Journal of Applied Probability*, 25(A):175–184.

Bellot, A., Branson, K., and van der Schaar, M. (2022). Neural graphical modelling in continuous-time: consistency guarantees and algorithms. In *International Conference on Learning Representations*.

Blom, T., Bongers, S., and Mooij, J. M. (2020). Beyond structural causal models: Causal constraints models. In *Uncertainty in Artificial Intelligence*, pages 585–594. PMLR.

Bongers, S., Blom, T., and Mooij, J. M. (2022). Causal modeling of dynamical systems. *arXiv preprint arXiv:1803.08784*.

Bongers, S., Forré, P., Peters, J., and Mooij, J. M. (2021). Foundations of structural causal models with cycles and latent variables. *The Annals of Statistics*, 49(5):2885–2915.

Bradbury, J., Frostig, R., Hawkins, P., Johnson, M. J., Leary, C., Maclaurin, D., Necula, G., Paszke, A., VanderPlas, J., Wanderman-Milne, S., and Zhang, Q. (2018). JAX: composable transformations of Python+NumPy programs.

Brouillard, P., Lachapelle, S., Lacoste, A., Lacoste-Julien, S., and Drouin, A. (2020). Differentiable causal discovery from interventional data. *Advances in Neural Information Processing Systems*, 33:21865–21877.

Chickering, D. M. (2003). Optimal structure identification with greedy search. *J. Mach. Learn. Res.*, 3:507–554.

Christmann, A. and Steinwart, I. (2010). Universal kernels on non-standard input spaces. *Advances in neural information processing systems*, 23.

Conley, T. G., Hansen, L. P., Luttmer, E. G., and Scheinkman, J. A. (1997). Short-term interest rates as subordinated diffusions. *The Review of Financial Studies*, 10(3):525–577.

Cox, P. M., Betts, R. A., Jones, C. D., Spall, S. A., and Totterdell, I. J. (2000). Acceleration of global warming due to carbon-cycle feedbacks in a coupled climate model. *Nature*, 408(6809):184–187.

Cuturi, M., Meng-Papaxanthos, L., Tian, Y., Bunne, C., Davis, G., and Teboul, O. (2022). Optimal transport tools (ott): A jax toolbox for all things wasserstein. *arXiv preprint arXiv:2201.12324*.

Dawid, A. P. (2010). Beware of the DAG! In *Causality: objectives and assessment*, pages 59–86. PMLR.

Dettling, P., Homs, R., Améndola, C., Drton, M., and Hansen, N. R. (2022). Identifiability in continuous Lyapunov models. *arXiv preprint arXiv:2209.03835*.

Dibaeinia, P. and Sinha, S. (2020). SERGIO: a single-cell expression simulator guided by gene regulatory networks. *Cell systems*, 11(3):252–271.

Duffie, D. and Glynn, P. (2004). Estimation of continuous-time Markov processes sampled at random time intervals. *Econometrica*, 72(6):1773–1808.

Eberhardt, F. and Scheines, R. (2007). Interventions and causal inference. *Philosophy of science*, 74(5):981–995.

Ethier, S. N. and Kurtz, T. G. (1986). *Markov processes: characterization and convergence*. John Wiley & Sons.

Geiger, D. and Heckerman, D. (1994). Learning Gaussian networks. In *Uncertainty Proceedings 1994*, pages 235–243. Elsevier.

Genevay, A., Peyré, G., and Cuturi, M. (2018). Learning generative models with sinkhorn divergences. In *International Conference on Artificial Intelligence and Statistics*, pages 1608–1617. PMLR.

Gorham, J. and Mackey, L. (2015). Measuring sample quality with Stein's method. *Advances in neural information processing systems*, 28.

Granger, C. W. J. (1969). Investigating causal relations by econometric models and cross-spectral methods. *Econometrica*, 37(3):424–438.

Gretton, A., Borgwardt, K. M., Rasch, M. J., Schölkopf, B., and Smola, A. (2012). A kernel two-sample test. *The Journal of Machine Learning Research*, 13(1):723–773.

Hansen, L. P. and Scheinkman, J. A. (1995). Back to the future: Generating moment implications for continuous-time Markov processes. *Econometrica*, 63(4):767–804.

Hansen, N. and Sokol, A. (2014). Causal interpretation of stochastic differential equations. *Electronic Journal of Probability*, 19:1–24.

Hasty, J., McMillen, D., and Collins, J. J. (2002). Engineered gene circuits. *Nature*, 420(6912):224–230.

Hauser, A. and Bühlmann, P. (2012). Characterization and greedy learning of interventional Markov equivalence classes of directed acyclic graphs. *The Journal of Machine Learning Research*, 13(1):2409–2464.

Hoel, E. P., Albantakis, L., and Tononi, G. (2013). Quantifying causal emergence shows that macro can beat micro. *Proceedings of the National Academy of Sciences*, 110(49):19790–19795.

Hyttinen, A., Eberhardt, F., and Hoyer, P. O. (2012). Learning linear cyclic causal models with latent variables. *The Journal of Machine Learning Research*, 13(1):3387–3439.

Hyvärinen, A. (2005). Estimation of non-normalized statistical models by score matching. *Journal of Machine Learning Research*, 6(4).

Hyvärinen, A., Zhang, K., Shimizu, S., and Hoyer, P. O. (2010). Estimation of a structural vector autoregression model using non-Gaussianity. *Journal of Machine Learning Research*, 11(5).

Immer, A., Schultheiss, C., Vogt, J. E., Schölkopf, B., Bühlmann, P., and Marx, A. (2023). On the identifiability and estimation of causal location-scale noise models. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 14316–14332. PMLR.

Jacobsen, M. (1993). A brief account of the theory of homogeneous Gaussian diffusions in finite dimensions. *Frontiers in Pure and Applied Probability 1*.

Kanagawa, M., Hennig, P., Sejdinovic, D., and Sriperumbudur, B. K. (2018). Gaussian processes and kernel methods: A review on connections and equivalences. *arXiv preprint arXiv:1807.02582*.

Khasminskii, R. (2011). *Stochastic stability of differential equations*, volume 66. Springer Science & Business Media.

Kolter, J. Z. and Manek, G. (2019). Learning stable deep dynamics models. *Advances in neural information processing systems*, 32.

Lacerda, G., Spirtes, P., Ramsey, J., and Hoyer, P. O. (2008). Discovering cyclic causal models by independent components analysis. In *Proceedings of the Twenty-Fourth Conference on Uncertainty in Artificial Intelligence*, UAI'08, page 366–374, Arlington, Virginia, USA. AUAI Press.

Liu, Q., Lee, J., and Jordan, M. (2016). A kernelized Stein discrepancy for goodness-of-fit tests. In *International conference on machine learning*, pages 276–284. PMLR.

Ljung, L. (1998). *System identification*. Springer.

Micchelli, C. A., Xu, Y., and Zhang, H. (2006). Universal kernels. *Journal of Machine Learning Research*, 7(12).

Mooij, J. M. and Heskes, T. (2013). Cyclic causal discovery from continuous equilibrium data. In *Proceedings of the Twenty-Ninth Conference on Uncertainty in Artificial Intelligence*, UAI'13, page 431–439, Arlington, Virginia, USA. AUAI Press.

Mooij, J. M., Janzing, D., Heskes, T., and Schölkopf, B. (2011). On causal discovery with cyclic additive noise models. *Advances in neural information processing systems*, 24.

Mooij, J. M., Janzing, D., and Schölkopf, B. (2013). From ordinary differential equations to structural causal models: The deterministic case. In *Proceedings of the Twenty-Ninth Conference on Uncertainty in Artificial Intelligence*, UAI'13, page 440–448, Arlington, Virginia, USA. AUAI Press.

Norman, T. M., Horlbeck, M. A., Replogle, J. M., Ge, A. Y., Xu, A., Jost, M., Gilbert, L. A., and Weissman, J. S. (2019). Exploring genetic interaction manifolds constructed from rich single-cell phenotypes. *Science*, 365(6455):786–793.

Øksendal, B. (2003). *Stochastic differential equations*. Springer.

Pamfil, R., Sriwattanaworachai, N., Desai, S., Pilgerstorfer, P., Georgatzis, K., Beaumont, P., and Aragam, B. (2020). DYNOTEARS: Structure learning from time-series data. In *International Conference on Artificial Intelligence and Statistics*, pages 1595–1605. PMLR.

Pearl, J. (2009). *Causality*. Cambridge university press.

Peters, J., Bauer, S., and Pfister, N. (2022). Causal models for dynamical systems. In *Probabilistic and*

Causal Inference: The Works of Judea Pearl, pages 671–690. Association for Computing Machinery.

Peters, J., Janzing, D., and Schölkopf, B. (2017). *Elements of causal inference: foundations and learning algorithms*. The MIT Press.

Peyré, G., Cuturi, M., et al. (2019). Computational optimal transport: With applications to data science. *Foundations and Trends® in Machine Learning*, 11(5-6):355–607.

Rasmussen, C. E. and Williams, C. K. (2006). *Gaussian processes for machine learning*, volume 1. Springer.

Richards, S. M., Berkenkamp, F., and Krause, A. (2018). The Lyapunov neural network: Adaptive stability certification for safe learning of dynamical systems. In *Conference on Robot Learning*, pages 466–476. PMLR.

Richardson, T. (1996). A polynomial-time algorithm for deciding Markov equivalence of directed cyclic graphical models. In *Proceedings of the Twelfth International Conference on Uncertainty in Artificial Intelligence*, UAI'96, page 462–469, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Rothenhäusler, D., Heinze, C., Peters, J., and Meinshausen, N. (2015). backShift: Learning causal cyclic graphs from unknown shift interventions. *Advances in Neural Information Processing Systems*, 28.

Rubenstein, P. K., Weichwald, S., Bongers, S., Mooij, J. M., Janzing, D., Grosse-Wentrup, M., and Schölkopf, B. (2017). Causal consistency of structural equation models. In *Proceedings of the 33rd Conference on Uncertainty in Artificial Intelligence (UAI)*, page ID 11.

Särkkä, S. and Solin, A. (2019). *Applied stochastic differential equations*, volume 10. Cambridge University Press.

Schölkopf, B. (2022). Causality for machine learning. In *Probabilistic and Causal Inference: The Works of Judea Pearl*, pages 765–804. Association for Computing Machinery.

Schölkopf, B. and Smola, A. J. (2002). *Learning with kernels*. MIT press.

Sethuraman, M. G., Lopez, R., Mohan, R., Fekri, F., Biancalani, T., and Hütter, J.-C. (2023). NODAGS-Flow: Nonlinear cyclic causal structure learning. In *International Conference on Artificial Intelligence and Statistics*, pages 6371–6387. PMLR.

Shimizu, S., Hoyer, P. O., Hyvärinen, A., Kerminen, A., and Jordan, M. (2006). A linear non-Gaussian acyclic model for causal discovery. *Journal of Machine Learning Research*, 7(10).

Smola, A., Gretton, A., Song, L., and Schölkopf, B. (2007). A Hilbert space embedding for distributions. In *Algorithmic Learning Theory: 18th International Conference, ALT 2007, Sendai, Japan, October 1-4, 2007. Proceedings 18*, pages 13–31. Springer.

Stein, C. (1972). A bound for the error in the normal approximation to the distribution of a sum of dependent random variables. In *Proceedings of the Sixth Berkeley Symposium on Mathematical Statistics and Probability, Volume 2: Probability Theory*, volume 6, pages 583–603. University of California Press.

Stein, M. L. (1999). *Interpolation of spatial data: some theory for kriging*. Springer Science & Business Media.

Steinwart, I. and Christmann, A. (2008). *Support vector machines*. Springer Science & Business Media.

Tank, A., Covert, I., Foti, N., Shojaie, A., and Fox, E. B. (2021). Neural Granger causality. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(8):4267–4279.

Tejada-Lapuerta, A., Bertin, P., Bauer, S., Aliee, H., Bengio, Y., and Theis, F. J. (2023). Causal machine learning for single-cell genomics. *arXiv preprint arXiv:2310.14935*.

Varando, G. and Hansen, N. R. (2020). Graphical continuous Lyapunov models. In *Conference on Uncertainty in Artificial Intelligence*, pages 989–998. PMLR.

Wang, Y., Solus, L., Yang, K., and Uhler, C. (2017). Permutation-based causal inference algorithms with interventions. *Advances in Neural Information Processing Systems*, 30.

Wendland, H. (2004). *Scattered data approximation*, volume 17. Cambridge university press.

Wong, E. (1964). The construction of a class of stationary Markoff processes. In *Proc. Sympos. Appl. Math., Vol. XVI*, pages 264–276.

Yuan, M. and Lin, Y. (2006). Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 68(1):49–67.

Zhang, J., Cammarata, L., Squires, C., Sapsis, T. P., and Uhler, C. (2022). Active learning for optimal intervention design in causal models. *arXiv preprint arXiv:2209.04744*.

Zhang, J., Squires, C., and Uhler, C. (2021). Matching a desired causal state via shift interventions. *Advances in Neural Information Processing Systems*, 34:19923–19934.

Zheng, X., Aragam, B., Ravikumar, P. K., and Xing, E. P. (2018). DAGs with NO TEARS: Continuous optimization for structure learning. *Advances in neural information processing systems*, 31.

Checklist

1. For all models and algorithms presented, check if you include:
   (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. — *These components are clearly defined and explained in Sections 2, 3, and 4.*
   (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. — *We formally analyze the properties of our algorithm in Section 3, but not the complexity, because it depends on the SDE model employed (see, e.g., Equations 3 or 6).*
   (c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. — *Our code is available at:* `https://github.com/larslorch/stadion`

2. For any theoretical claim, check if you include:
   (a) Statements of the full set of assumptions of all theoretical results. — *All results in Section 3 state their assumptions. We state any assumptions that hold throughout the paper, like Lipschitz continuity of the SDE functions, in Section 2.*
   (b) Complete proofs of all theoretical results. — *We provide proof sketches of the theoretical results in Section 3 and complete proofs in Appendix B.*
   (c) Clear explanations of any assumptions. — *Explanations of any non-standard assumptions are given in Sections 2 and 3 or Appendix B.*

3. For all figures and tables that present empirical results, check if you include:
   (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). — *See code link provided above.*
   (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). — *Inference and experimental details are summarized in Section 6 and Appendix D.*
   (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). — *Figures 4 and 6 and Table 2 define the statistics and error bars used.*
   (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). — *Appendix D.6 describes the compute infrastructure used for the experiments.*

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:
   (a) Citations of the creator If your work uses existing assets. — *Appendix D cites or links all existing code and data assets used for the experiments as well as their licences.*
   (b) The license information of the assets, if applicable. — *See above.*
   (c) New assets either in the supplemental material or as a URL, if applicable. — *See code link provided above.*
   (d) Information about consent from data providers/curators. — *Not applicable.*
   (e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. — *Not applicable.*

5. If you used crowdsourcing or conducted research with human subjects, check if you include:
   (a) The full text of instructions given to participants and screenshots. — *Not applicable.*
   (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. — *Not applicable.*
   (c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. — *Not applicable.*

A Additional Background

A.1 Euler-Maruyama Method

To approximate the solutions to SDEs, we use the Euler-Maruyama method (Särkkä and Solin, 2019, Section 8.2). The Euler-Maruyama approximation of sample paths of the diffusion $\{\mathbf{x}_t\}$ solving $\mathrm{d}\mathbf{x}_t = f(\mathbf{x}_t)\mathrm{d}t + \sigma(\mathbf{x}_t)\mathrm{d}\mathbb{W}_t$ is given by

$$\mathbf{x}_{l+1} := \mathbf{x}_l + f(\mathbf{x}_l)\Delta t + \sigma(\mathbf{x}_l)\boldsymbol{\xi}_l\sqrt{\Delta t} \tag{9}$$

for some step size Δt and independent vectors $\boldsymbol{\xi}_l \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. To generate L samples from the stationary density $\mu(\mathbf{x})$ of $\{\mathbf{x}_t\}$, we simulate a single sample path and then select every k-th state $\mathbf{x}_{l\cdot k}$ for $l \in \{1, \ldots, L\}$ as a sample, where k is a thinning factor as in Markov chain Monte Carlo. In our experiments, we sample $\mathbf{x}_0 \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and use a step size of $\Delta t = 0.01$, a thinning factor of 500, and 100 samples of burn-in, which we selected based on autocorrelation diagnostics of the thinned Markov chains.

A.2 Sobolev Spaces

Some of our theoretical results build on the notion of Sobolev spaces. While not required here, we recommend Adams and Fournier (2003) for a detailed introduction. The Sobolev norm $\|\cdot\|_{m,p}$ of a function f sums the L_p norms of all its partial derivatives up to order m and is defined as

$$\|f\|_{m,p} := \left(\sum_{\mathbf{n} \in \mathbb{N}_0^d : |\mathbf{n}| \leq m} \left\| \frac{\partial^{n_1}}{\partial x_1^{n_1}} \cdots \frac{\partial^{n_d}}{\partial x_d^{n_d}} f \right\|_p^p \right)^{1/p}$$

for $1 \leq p < \infty$. Here, $\|\cdot\|_p$ is the L_p norm defined as $\|f\|_p = \left(\int_{\mathbb{R}^d} |f(\mathbf{x})|^p \mathrm{d}\mathbf{x}\right)^{1/p}$. The Sobolev space $W^{m,p}$ contains all functions $f : \mathbb{R}^d \to \mathbb{R}$ such that $\|f\|_{m,p} < \infty$. Moreover, the space $W_c^{m,p}$ is defined as the closure of C_c^∞ in $W^{m,p}$ (Adams and Fournier, 2003, Section 3.2).

A.3 Matérn Kernel

The Matérn kernel $k_{\nu,\gamma}$ with smoothness and scale parameters $\nu, \sigma > 0$ can be seen as a generalization of the Gaussian kernel that allows controlling the smoothness of the RKHS functions. We write the Matérn kernel $k_{\nu,\gamma}(\mathbf{x}, \mathbf{x}')$ in terms of the distance $r = \|\mathbf{x} - \mathbf{x}'\|_2$ as

$$k_{\nu,\gamma}(r) := \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\frac{\sqrt{2\nu}\, r}{\gamma} \right)^\nu K_\nu \left(\frac{\sqrt{2\nu}\, r}{\gamma} \right), \tag{10}$$

where Γ is the gamma function and K_ν is a modified Bessel function of the second kind and order ν (Rasmussen and Williams, 2006, Equation 4.14). Common special cases of $k_{\nu,\gamma}$ have the following explicit forms:

$$\begin{aligned}
k_{\nu=1/2,\gamma}(r) &= \exp\left(-\frac{r}{\gamma}\right) \\
k_{\nu=3/2,\gamma}(r) &= \left(1 + \frac{\sqrt{3}r}{\gamma}\right) \exp\left(-\frac{\sqrt{3}r}{\gamma}\right) \\
k_{\nu=5/2,\gamma}(r) &= \left(1 + \frac{\sqrt{5}r}{\gamma} + \frac{5r^2}{3\gamma^2}\right) \exp\left(-\frac{\sqrt{5}r}{\gamma}\right)
\end{aligned}$$

The Gaussian kernel $k_\gamma(r) = \exp(-r^2/2\gamma^2)$ is obtained from $k_{\nu,\gamma}$ as $\nu \to \infty$.

The following two results will be useful for proving Theorem 3. The first statement was originally shown by Wendland (2004, Corollary 10.48) and follows from Rasmussen and Williams (2006, Equation 4.15), linking the Matérn RKHS to the Sobolev spaces. The second result concerns the differentiability of the Matérn kernel function:

Lemma 4 (Kanagawa et al., 2018, Example 2.8) *The RKHS $\mathcal{H}$ of a Matérn kernel $k_{\nu,\gamma}$ is norm-equivalent to the Sobolev space $W^{\nu+d/2,2}$. Specifically, we have $h \in \mathcal{H}$ if and only if $h \in W^{\nu+d/2,2}$. Moreover, there exist constants c_1, c_2 such that $c_1\|h\|_{\nu+d/2,2} \leq \|h\|_{\mathcal{H}} \leq c_2\|h\|_{\nu+d/2,2}$ for all $h \in \mathcal{H}$.*

Lemma 5 (Stein, 1999, Section 2.7, p. 32) *The Matérn covariance function $k_{\nu,\gamma}(r)$ is $2k$-times differentiable if and only if $\nu > k$.*

B Proofs

B.1 Proof of Lemma 1

Let $\mathcal{B}_{\mu,\mathcal{L}} : \mathcal{H} \to \mathbb{R}$ be the composition of the operators $\mathbb{E}_{\mathbf{x}\sim\mu}$ and $\mathcal{L}$ defined as $\mathcal{B}_{\mu,\mathcal{L}}h := \mathbb{E}_{\mathbf{x}\sim\mu}[\mathcal{L}h(\mathbf{x})]$ for all $h \in \mathcal{H}$. The functional $\mathcal{B}_{\mu,\mathcal{L}}$ is linear since both the expectation and $\mathcal{L}$ are linear operators. Moreover, $\mathcal{B}_{\mu,\mathcal{L}}$ is continuous because the functional is bounded on the unit ball of functions in $\mathcal{H}$:

$$
\begin{aligned}
|\mathcal{B}_{\mu,\mathcal{L}}h| &= |\mathbb{E}_{\mathbf{x}\sim\mu}[\mathcal{L}h(\mathbf{x})]| \\
&\leq \mathbb{E}_{\mathbf{x}\sim\mu}[|\mathcal{L}h(\mathbf{x})|] && \text{Jensen's inequality} \\
&= \mathbb{E}_{\mathbf{x}\sim\mu}\left[\left|f(\mathbf{x})\cdot\nabla_{\mathbf{x}}h(\mathbf{x}) + \tfrac{1}{2}\operatorname{tr}(\sigma(\mathbf{x})\sigma(\mathbf{x})^\top\nabla_{\mathbf{x}}\nabla_{\mathbf{x}}h(\mathbf{x}))\right|\right] && \text{by definition} \\
&\leq \mathbb{E}_{\mathbf{x}\sim\mu}\left[\left|f(\mathbf{x})\cdot\nabla_{\mathbf{x}}h(\mathbf{x})\right| + \tfrac{1}{2}\left|\operatorname{tr}(\sigma(\mathbf{x})\sigma(\mathbf{x})^\top\nabla_{\mathbf{x}}\nabla_{\mathbf{x}}h(\mathbf{x}))\right|\right] && \text{triangle inequality} \\
&\leq \mathbb{E}_{\mathbf{x}\sim\mu}\left[\|f(\mathbf{x})\|_2\|\nabla_{\mathbf{x}}h(\mathbf{x})\|_2 + \tfrac{1}{2}\|\sigma(\mathbf{x})\sigma(\mathbf{x})^\top\|_{\mathrm{F}}\|\nabla_{\mathbf{x}}\nabla_{\mathbf{x}}h(\mathbf{x})\|_{\mathrm{F}}\right] && \text{Cauchy-Schwarz} \\
&= \mathbb{E}_{\mathbf{x}\sim\mu}\left[\|f(\mathbf{x})\|_2\Big(\sum_{i=1}^{d}\left|\tfrac{\partial}{\partial x_i}h(\mathbf{x})\right|^2\Big)^{1/2} + \tfrac{1}{2}\|\sigma(\mathbf{x})\sigma(\mathbf{x})^\top\|_{\mathrm{F}}\Big(\sum_{i=1}^{d}\sum_{j=1}^{d}\left|\tfrac{\partial^2}{\partial x_i\partial x_j}h(\mathbf{x})\right|^2\Big)^{1/2}\right] \\
&\leq \|h\|_{\mathcal{H}}\cdot\mathbb{E}_{\mathbf{x}\sim\mu}\left[\|f(\mathbf{x})\|_2\Big(\sum_{i=1}^{d}\tfrac{\partial}{\partial x_{i,i}}k(\mathbf{x},\mathbf{x})\Big)^{1/2} + \tfrac{1}{2}\|\sigma(\mathbf{x})\sigma(\mathbf{x})^\top\|_{\mathrm{F}}\Big(\sum_{i=1}^{d}\sum_{j=1}^{d}\tfrac{\partial^2}{\partial x_{i,i}\partial x_{j,j}}k(\mathbf{x},\mathbf{x})\Big)^{1/2}\right]. \quad (*)
\end{aligned}
$$

The last inequality follows from the fact that the partial derivatives of RKHS functions $h \in \mathcal{H}$ are bounded as

$$
\left|\tfrac{\partial}{\partial x_i}h(\mathbf{x})\right| \leq \|h\|_{\mathcal{H}}\cdot\left(\tfrac{\partial}{\partial x_{i,i}}k(\mathbf{x},\mathbf{x})\right)^{1/2} \qquad \text{and} \qquad \left|\tfrac{\partial^2}{\partial x_i\partial x_j}h(\mathbf{x})\right| \leq \|h\|_{\mathcal{H}}\cdot\left(\tfrac{\partial^2}{\partial x_{i,i}\partial x_{j,j}}k(\mathbf{x},\mathbf{x})\right)^{1/2}
$$

(Steinwart and Christmann, 2008, Corollary 4.36). The notation $\partial/\partial x_{i,i}$ was defined in Footnote 2. By assumption, the squares of f, σ, and the partial derivatives $\partial/\partial x_{i,i}k(\mathbf{x},\mathbf{x})$ and $\partial^2/\partial x_{i,i}\partial x_{j,j}k(\mathbf{x},\mathbf{x})$ are square-integrable with respect to μ, hence the expectation in the above inequality is bounded. Thus, when applied to the unit ball of $\mathcal{H}$, for which $\|h\|_{\mathcal{H}} \leq 1$, the norm of $\mathcal{B}_{\mu,\mathcal{L}}$ is bounded. Hence, $\mathcal{B}_{\mu,\mathcal{L}}$ is a continuous linear functional, and by the Riesz representation theorem, there exists a unique representer $g_{\mu,\mathcal{L}} \in \mathcal{H}$ such that

$$
\mathcal{B}_{\mu,\mathcal{L}}h = \mathbb{E}_{\mathbf{x}\sim\mu}[\mathcal{L}_{\mathbf{x}}h(\mathbf{x})] = \langle h, g_{\mu,\mathcal{L}}\rangle_{\mathcal{H}} \tag{11}
$$

for all $h \in \mathcal{H}$. We obtain the explicit form for $g_{\mu,\mathcal{L}}$ by substituting $k(\,\cdot\,,\mathbf{x}') \in \mathcal{H}$ for h in (11), which yields

$$
\mathbb{E}_{\mathbf{x}\sim\mu}[\mathcal{L}_{\mathbf{x}}k(\mathbf{x},\mathbf{x}')] = \langle k(\,\cdot\,,\mathbf{x}'), g_{\mu,\mathcal{L}}\rangle_{\mathcal{H}}\,,
$$

where the notation $\mathcal{L}_{\mathbf{x}}$ makes explicit that the operator $\mathcal{L}$ is applied to the argument $\mathbf{x}$. By the reproducing property, we have $\langle k(\,\cdot\,,\mathbf{x}'), g_{\mu,\mathcal{L}}\rangle_{\mathcal{H}} = g_{\mu,\mathcal{L}}(\mathbf{x}')$ (Schölkopf and Smola, 2002). Hence, $g_{\mu,\mathcal{L}}(\mathbf{x}') = \mathbb{E}_{\mathbf{x}\sim\mu}[\mathcal{L}_{\mathbf{x}}k(\mathbf{x},\mathbf{x}')]$, and we can read off that the representer function is given by $g_{\mu,\mathcal{L}}(\cdot) = \mathbb{E}_{\mathbf{x}\sim\mu}[\mathcal{L}_{\mathbf{x}}k(\mathbf{x},\,\cdot\,)]$. ■

B.2 Proof of Theorem 2

Using the representer function $g_{\mu,\mathcal{L}}$ in Lemma 1, we can express the supremum over $\mathcal{F}$ as

$$
\sup_{h\in\mathcal{F}}\mathbb{E}_{\mathbf{x}\sim\mu}\big[\mathcal{L}h(\mathbf{x})\big] = \sup_{h\in\mathcal{F}}\langle h, g_{\mu,\mathcal{L}}\rangle_{\mathcal{H}} = \Big\langle \tfrac{g_{\mu,\mathcal{L}}}{\|g_{\mu,\mathcal{L}}\|_{\mathcal{H}}}, g_{\mu,\mathcal{L}}\Big\rangle_{\mathcal{H}} = \|g_{\mu,\mathcal{L}}\|_{\mathcal{H}}\,.
$$

In the above, we used the fact that the norm $\langle h, g_{\mu,\mathcal{L}}\rangle_{\mathcal{H}}$ is maximized over $h \in \mathcal{F}$ by the unit-norm function aligned with $g_{\mu,\mathcal{L}}$, that is, by $g_{\mu,\mathcal{L}}/\|g_{\mu,\mathcal{L}}\|_{\mathcal{H}} \in \mathcal{F}$. The squared RKHS norm $\|g_{\mu,\mathcal{L}}\|_{\mathcal{H}}^2$ can be written in terms of the kernel as

$$
\begin{aligned}
\|g_{\mu,\mathcal{L}}\|_{\mathcal{H}}^2 &= \langle g_{\mu,\mathcal{L}}, g_{\mu,\mathcal{L}}\rangle_{\mathcal{H}} = \mathbb{E}_{\mathbf{x}\sim\mu}\big[\mathcal{L}_{\mathbf{x}}g_{\mu,\mathcal{L}}(\mathbf{x})\big] && \text{Lemma 1} \\
&= \mathbb{E}_{\mathbf{x}\sim\mu}\big[\mathcal{L}_{\mathbf{x}}\big[\mathbb{E}_{\mathbf{x}'\sim\mu}\big[\mathcal{L}_{\mathbf{x}'}k(\mathbf{x}',\mathbf{x})\big]\big]\big]\,. && \text{Explicit form of } g_{\mu,\mathcal{L}}(\cdot)
\end{aligned}
$$

We call this expression the *kernel deviation from stationarity* $\mathrm{KDS}(\mathcal{L},\mu;\mathcal{F})$.

Under additional regularity conditions on f, σ, k, and μ, we may interchange the differentials in $\mathcal{L}_{\mathbf{x}}$ with the integral in $\int\mu(\mathbf{x}')\mathcal{L}_{\mathbf{x}'}k(\mathbf{x}',\mathbf{x})\mathrm{d}\mathbf{x}'$ (or specifically, their involved limits) and write $\mathrm{KDS}(\mathcal{L},\mu;\mathcal{F}) = \mathbb{E}_{\mathbf{x}\sim\mu,\mathbf{x}'\sim\mu}\big[\mathcal{L}_{\mathbf{x}}\mathcal{L}_{\mathbf{x}'}k(\mathbf{x},\mathbf{x}')\big]$. For example, by the dominated convergence theorem, one sufficient condition allowing the interchange is when the functions and their first- and second-order partial derivatives are continuous and bounded. More general conditions are possible. ■

B.3 Proof of Theorem 3

Let $\mathcal{H}$ be the RKHS of the Matérn kernel $k_{\nu,\gamma}$, and let $\mathcal{F}$ be its unit ball. This proof uses Lemmata 4 and 5, two auxiliary results about Matérn and Sobolev spaces that are given in Appendix A.

To begin, we note that f, σ, $\partial/\partial x_{i,i} k_{\nu,\gamma}(\mathbf{x},\mathbf{x})$, and $\partial^2/\partial x_{i,i}\partial x_{j,j} k_{\nu,\gamma}(\mathbf{x},\mathbf{x})$ are all square-integrable with respect to μ, because the functions are bounded, and any bounded function is square-integrable with respect to a probability density. Both functions f and σ are bounded by assumption. Moreover, Lemma 5 and $\nu > 2$ imply that the partial derivatives $\partial/\partial x_{i,i} k_{\nu,\gamma}(\mathbf{x},\mathbf{x})$ and $\partial^2/\partial x_{i,i}\partial x_{j,j} k_{\nu,\gamma}(\mathbf{x},\mathbf{x})$ exist and are finite. These functions of $\mathbf{x}$ are bounded, because the Matérn kernel function depends only on the distance between its inputs, which is $\|\mathbf{x}-\mathbf{x}\|_2 = 0$ for any $\mathbf{x}$, and thus these partial derivatives are constant with respect to $\mathbf{x}$. Given the square-integrability of f, σ, $\partial/\partial x_{i,i} k_{\nu,\gamma}(\mathbf{x},\mathbf{x})$, and $\partial^2/\partial x_{i,i}\partial x_{j,j} k_{\nu,\gamma}(\mathbf{x},\mathbf{x})$, all assumptions of Lemma 1 and Theorem 2 are satisfied.

To prove the theorem, we leverage the fact that the smooth functions with compact support C_c^∞ form a core for the generator $\mathcal{A}$ associated to the SDEs when f, σ are Lipschitz continuous and bounded and the matrix $\sigma(\mathbf{x})\sigma(\mathbf{x})^\top$ is positive definite for all $\mathbf{x} \in \mathbb{R}^d$ (Ethier and Kurtz, 1986, Theorem 1.6, p. 370). We can link the core C_c^∞ to the Matérn RKHS $\mathcal{H}$:

Lemma 6 *C_c^∞ is a dense subset of $\mathcal{H}$ with respect to the Sobolev norm $\|\cdot\|_{\nu+d/2,2}$.*

Proof of Lemma 6. The space $W_c^{m,p}$ is defined as the closure of C_c^∞ in the Sobolev space $W^{m,p}$ (Appendix A.2). Therefore, the core C_c^∞ is dense in $W_c^{m,p}$ with respect to the Sobolev norm $\|\cdot\|_{m,p}$. Moreover, $W_c^{m,p} = W^{m,p}$ when both spaces are defined over $\mathbb{R}^d$ (Adams and Fournier, 2003, Corollary 3.23), so C_c^∞ is dense in $W^{m,p}$. From Lemma 4, we know that $W^{\nu+d/2,2} = \mathcal{H}$ for the set of functions. Hence, C_c^∞ is dense in $W^{\nu+d/2,2} = \mathcal{H}$ with respect to the Sobolev norm $\|\cdot\|_{\nu+d/2,2}$.

We now prove both directions of the equivalence in the theorem:

$\Leftarrow$ If $\mathrm{KDS}(\mathcal{L}, \mu; \mathcal{F}) = 0$, then $\sup_{h\in\mathcal{F}} \mathbb{E}_{\mathbf{x}\sim\mu}[\mathcal{L}h(\mathbf{x})] = 0$ by Theorem 2. Since the supremum is nonnegative, it follows that $\mathbb{E}_{\mathbf{x}\sim\mu}[\mathcal{L}h(\mathbf{x})] = 0$ for all $h \in \mathcal{F}$. This implies that the equality also holds for $h \in \mathcal{H}$, since the length of the vectors does not affect their orthogonality. When $\|h\|_\mathcal{H} > 0$, we can also see this from $\mathbb{E}_{\mathbf{x}\sim\mu}[\mathcal{L}h(\mathbf{x})] = \langle h, g_{\mu,\mathcal{L}}\rangle_\mathcal{H} = \|h\|_\mathcal{H}\langle h/\|h\|_\mathcal{H}, g_{\mu,\mathcal{L}}\rangle_\mathcal{H} = \|h\|_\mathcal{H} \cdot 0 = 0$ since $h/\|h\|_\mathcal{H} \in \mathcal{F}$.

By Lemma 6, the core C_c^∞ is a subset of $\mathcal{H}$, so we have $\mathbb{E}_{\mathbf{x}\sim\mu}[\mathcal{L}u(\mathbf{x})] = 0$ for all $u \in C_c^\infty$. If $u \in C_c^\infty$, then $u \in C_c^2$ and thus $\mathcal{A}u = \mathcal{L}u$. It follows that $\mathbb{E}_{\mathbf{x}\sim\mu}[\mathcal{A}u(\mathbf{x})] = 0$ for all u in the core C_c^∞. This implies that μ is the stationary density (Ethier and Kurtz, 1986, Chapter 4, Proposition 9.2).

$\Rightarrow$ If μ is the stationary density, we have $\mathbb{E}_{\mathbf{x}\sim\mu}[\mathcal{A}u(\mathbf{x})] = 0$ for all functions u in the core C_c^∞. Moreover, since $C_c^\infty \subset C_c^2$, it holds that $\mathbb{E}_{\mathbf{x}\sim\mu}[\mathcal{L}u(\mathbf{x})] = 0$.

Let $h \in \mathcal{H}$. By Lemma 6, there exists $u \in C_c^\infty$ such that $\|h-u\|_{\nu+d/2,2} < \epsilon$. By the above, we then have

$$
\begin{aligned}
|\mathbb{E}_{\mathbf{x}\sim\mu}[\mathcal{L}h(\mathbf{x})]| &= |\mathbb{E}_{\mathbf{x}\sim\mu}[\mathcal{L}h(\mathbf{x}) - \mathcal{L}u(\mathbf{x}) + \mathcal{L}u(\mathbf{x})]| && \text{expanding} \\
&\le |\mathbb{E}_{\mathbf{x}\sim\mu}[\mathcal{L}(h-u)(\mathbf{x})]| + |\mathbb{E}_{\mathbf{x}\sim\mu}[\mathcal{L}u(\mathbf{x})]| && \text{triangle inequality} \\
&= |\mathbb{E}_{\mathbf{x}\sim\mu}[\mathcal{L}(h-u)(\mathbf{x})]| \\
&= |\langle h-u, g_{\mu,\mathcal{L}}\rangle_\mathcal{H}| && \text{Lemma 1} \\
&\le \|h-u\|_\mathcal{H}\, \|g_{\mu,\mathcal{L}}\|_\mathcal{H} && \text{Cauchy-Schwarz} \\
&\le c_2 \|h-u\|_{\nu+d/2,2}\, \|g_{\mu,\mathcal{L}}\|_\mathcal{H} && \text{Lemma 4} \\
&< c_2\, \epsilon\, \|g_{\mu,\mathcal{L}}\|_\mathcal{H}
\end{aligned}
$$

Thus, $|\mathbb{E}_{\mathbf{x}\sim\mu}[\mathcal{L}h(\mathbf{x})]|$ is bounded by ϵ times the constants c_2 and $\|g_{\mu,\mathcal{L}}\|_\mathcal{H}$, which are both independent of the function h. Hence, for all functions $h \in \mathcal{H}$ and any $\epsilon' > 0$, we can choose $\epsilon > 0$ such that $|\mathbb{E}_{\mathbf{x}\sim\mu}[\mathcal{L}h(\mathbf{x})]| < \epsilon'$. It follows that $\mathbb{E}_{\mathbf{x}\sim\mu}[\mathcal{L}h(\mathbf{x})] = 0$ for all $h \in \mathcal{H}$ and, by Theorem 2, $\mathrm{KDS}(\mathcal{L}, \mu; \mathcal{F}) = 0$.

■

```python
import grad, hessian, trace

f = ...       # function: [d], theta -> [d]
sigma = ...   # function: [d], theta -> [d, d]
kernel = ...  # function: [d], [d], theta -> []

def L(h, arg):
  # apply operator L to h(x, y, theta) at position 'arg'
  def Lh(x, y, theta):
    z = x if arg == 0 else y
    v = f(z, theta)
    m = sigma(z, theta)
    return v @ grad(h, arg=arg)(x, y, theta) \
           + 0.5 * trace(m @ m.T @ hessian(h, arg=arg)(x, y, theta))
  return Lh

# function: [d], [d], theta -> []
kds = L(L(kernel, arg=0), arg=1)

# function: [d], [d], theta -> theta
dkds_dtheta = grad(kds, arg=2)
```

Figure 5: Computing the generator gradient $\nabla_{\boldsymbol{\theta}} \mathcal{L}_{\mathbf{x}}^{\boldsymbol{\theta}} \mathcal{L}_{\mathbf{x}'}^{\boldsymbol{\theta}} k(\mathbf{x}, \mathbf{x}')$ via two calls of the operator $\mathcal{L}^{\boldsymbol{\theta}}$ (here: `L`).

C Additional Details on the Kernel Deviation from Stationarity

C.1 Linear-Time Unbiased Estimator

The empirical estimate of $\mathrm{KDS}(\mathcal{L}, \mu; \mathcal{F})$ in (5) scales quadratically with the size of $D = \{\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(N)}\}$. Similar to Gretton et al. (2012), we can obtain an alternative estimator by subsampling the summands in (5) as

$$\hat{\mathrm{KDS}}_{\mathrm{linear}}(\mathcal{L}, D; k) := \frac{1}{\lfloor N/2 \rfloor} \sum_{m=1}^{\lfloor N/2 \rfloor} \mathcal{L}_{\mathbf{x}} \mathcal{L}_{\mathbf{x}'} k(\mathbf{x}^{(2m-1)}, \mathbf{x}^{(2m)}) \,. \tag{12}$$

The computation of this estimator scales linearly with N. The estimator is unbiased by the linearity of expectations and the fact that samples are i.i.d. as μ. The linear estimator may be advantageous when N is prohibitively large to compute (5) or when a single evaluation of (the gradient of) $\mathcal{L}_{\mathbf{x}}^{\boldsymbol{\theta}} \mathcal{L}_{\mathbf{x}'}^{\boldsymbol{\theta}} k(\mathbf{x}, \mathbf{x}')$ is expensive, but we do not want to ignore samples from D.

C.2 General Explicit Form and Automatic Differentiation

For general diffusion functions σ, we cannot use the Laplacian notation of (6), but $\mathcal{L}_{\mathbf{x}}^{\boldsymbol{\theta}} \mathcal{L}_{\mathbf{x}'}^{\boldsymbol{\theta}} k(\mathbf{x}, \mathbf{x}')$ nevertheless has an explicit form given by

$$\begin{aligned} \mathcal{L}_{\mathbf{x}}^{\boldsymbol{\theta}} \mathcal{L}_{\mathbf{x}'}^{\boldsymbol{\theta}} k(\mathbf{x}, \mathbf{x}') = {} & f_{\boldsymbol{\theta}}(\mathbf{x}) \cdot \nabla_{\mathbf{x}} \nabla_{\mathbf{x}'} k(\mathbf{x}, \mathbf{x}') \cdot f_{\boldsymbol{\theta}}(\mathbf{x}') \\ & + \tfrac{1}{2} f_{\boldsymbol{\theta}}(\mathbf{x}) \cdot \nabla_{\mathbf{x}} \operatorname{tr}\left(\sigma_{\boldsymbol{\theta}}(\mathbf{x}) \sigma_{\boldsymbol{\theta}}(\mathbf{x})^\top \nabla_{\mathbf{x}'} \nabla_{\mathbf{x}'} k(\mathbf{x}, \mathbf{x}')\right) \\ & + \tfrac{1}{2} f_{\boldsymbol{\theta}}(\mathbf{x}') \cdot \nabla_{\mathbf{x}'} \operatorname{tr}\left(\sigma_{\boldsymbol{\theta}}(\mathbf{x}') \sigma_{\boldsymbol{\theta}}(\mathbf{x}')^\top \nabla_{\mathbf{x}} \nabla_{\mathbf{x}} k(\mathbf{x}, \mathbf{x}')\right) \\ & + \tfrac{1}{4} \operatorname{tr}\Big(\sigma_{\boldsymbol{\theta}}(\mathbf{x}) \sigma_{\boldsymbol{\theta}}(\mathbf{x})^\top \nabla_{\mathbf{x}} \nabla_{\mathbf{x}} \operatorname{tr}\left(\sigma_{\boldsymbol{\theta}}(\mathbf{x}') \sigma_{\boldsymbol{\theta}}(\mathbf{x}')^\top \nabla_{\mathbf{x}'} \nabla_{\mathbf{x}'} k(\mathbf{x}, \mathbf{x}')\right)\Big) \,. \end{aligned} \tag{13}$$

Precomputing the kernel terms is still possible to some degree, but fewer steps of the computation can be cached in the trace terms. In this case, it may thus be simpler implementation-wise to compute the parameter gradients $\nabla_{\boldsymbol{\theta}} \mathcal{L}_{\mathbf{x}}^{\boldsymbol{\theta}} \mathcal{L}_{\mathbf{x}'}^{\boldsymbol{\theta}} k(\mathbf{x}, \mathbf{x}')$ and $\nabla_{\boldsymbol{\phi}} \mathcal{L}_{\mathbf{x}}^{\boldsymbol{\theta}, \boldsymbol{\phi}} \mathcal{L}_{\mathbf{x}'}^{\boldsymbol{\theta}, \boldsymbol{\phi}} k(\mathbf{x}, \mathbf{x}')$ with automatic differentiation at each update step in Algorithm 1.

To make this concrete, Figure 5 provides Python pseudocode using `autograd` and `JAX`-style syntax (Bradbury et al., 2018) that illustrates how to compute $\nabla_{\boldsymbol{\theta}} \mathcal{L}_{\mathbf{x}}^{\boldsymbol{\theta}} \mathcal{L}_{\mathbf{x}'}^{\boldsymbol{\theta}} k(\mathbf{x}, \mathbf{x}')$ in only a few lines. Following (3), the operator $\mathcal{L}^{\boldsymbol{\theta}}$ can be conveniently defined as a higher-order function that accepts a function h as input and returns the function $\mathcal{L}^{\boldsymbol{\theta}} h$. After applying $\mathcal{L}^{\boldsymbol{\theta}}$ once to each argument of $k(\mathbf{x}, \mathbf{x}')$, the function `kds` in Figure 5 exactly computes (13), and we can directly compute its gradient with respect to $\boldsymbol{\theta}$ in the last line using automatic differentiation.

D Experimental Setup

D.1 Data

D.1.1 Causal Structures

For benchmarking, we simulate data from randomly-generated sparse linear systems and sparse gene regulatory network models with $d = 20$ variables. Following prior work (e.g., Zheng et al., 2018), we sample random causal structures $\mathbf{G} \in \{0,1\}^{d\times d}$ with the number of causal dependencies per variable following a polynomial or power-law distribution, corresponding to Erdős-Rényi and scale-free graphs, respectively. Erdős-Rényi graphs are sampled by drawing links independently with a fixed probability (when acyclic, restricted to an upper-triangular matrix). Scale-free graphs are generated by preferential attachment, where links j to the previous $j-1$ nodes are sampled with probability proportional to its degree and then randomly directed (when acyclic, always directed ingoing to j). For both structure distributions, we fix the expected degree of the variables to 3.

D.1.2 Cyclic Linear Systems

Models Given some $\mathbf{G} \in \{0,1\}^{d\times d}$, we generate random instances of the two cyclic linear models

$$\mathbf{x} = \mathbf{W}\mathbf{x} + \mathbf{b} + \operatorname{diag}(\boldsymbol{\sigma})\boldsymbol{\epsilon} \quad \text{with } \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}) \tag{Cyclic Linear SCM}$$

$$\mathrm{d}\mathbf{x}_t = (\mathbf{W}\mathbf{x}_t + \mathbf{b})\mathrm{d}t + \operatorname{diag}(\boldsymbol{\sigma})\mathrm{d}\mathbb{W}_t \tag{Cyclic Linear SDE}$$

where $\mathbf{W} \in \mathbb{R}^{d\times d}$, $\mathbf{b} \in \mathbb{R}^d$, and $\boldsymbol{\sigma} \in \mathbb{R}^d_{>0}$, and $\mathbf{W}$ is sparse according to $\mathbf{G}$. Sampling cyclic systems requires more caution than in the acyclic case, since both generative processes must be stable. For SCMs, the maximum of the real-parts of the eigenvalues $\rho(\mathbf{W})$ must be less than 1, for SDEs less than 0. For an insightful evaluation, we additionally want $\mathbf{W}$ to be asymmetric and not approximately diagonal, i.e., have significant causal dependencies between the variables.

To generate such systems, we first sample $\mathbf{G} \sim p(\mathbf{G})$, $\mathbf{W} \sim p(\mathbf{W})$, $\mathbf{b} \sim p(\mathbf{b})$, $\boldsymbol{\sigma} \sim p(\boldsymbol{\sigma})$. Then, we multiply $\mathbf{W}$ times $\mathbf{G}$ elementwise along their offdiagonal elements and finally subtract $\rho(\mathbf{W}) + \epsilon$ from the diagonal of $\mathbf{W}$, which ensures that $\rho(\mathbf{W}) \leq -\epsilon$. This protocol empirically induced stronger variable correlations in the stationary distributions than the procedure by Varando and Hansen (2020). They perform a more vacuous diagonal shift based on the Gershgorin circle theorem, often resulting in large dominating diagonals. For our experiments, we use $p(w_{ij}) = \mathrm{Unif}(-3,-1) \cup (1,3)$ and $\epsilon = 0.5$ for the matrices and $p(b_j) = \mathrm{Unif}(-3,3)$ and $p(\log \sigma_j) = \mathrm{Unif}(-1,1)$ for the biases and scales, respectively, both for the SCMs and SDEs. To sample the SCM data, we draw $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and then compute $\mathbf{x} = (\mathbf{I} - \mathbf{W})^{-1}(\mathbf{b} + \operatorname{diag}(\boldsymbol{\sigma})\boldsymbol{\epsilon})$ (Hyttinen et al., 2012). To sample from the stationary density of the SDEs, we use the Euler-Maruyama scheme (Appendix A.1).

Interventions Given the fully-specified linear model, we sample an observational dataset and interventional data for single-variable shift interventions on all variables, each with 1000 observations, as the interventions for the benchmark. In both SCMs and SDEs, the shift intervention is implemented by adding a scalar δ to the bias b_j of the target variable j. In our experiments, we sample $\delta \sim p(\delta)$ with $p(\delta) = \mathrm{Unif}(-15,-5) \cup (5,15)$ independently for each intervention.

D.1.3 Gene Regulatory Networks

Model Given some acyclic $\mathbf{G} \in \{0,1\}^{d\times d}$, we use the SERGIO model by Dibaeinia and Sinha (2020) and their corresponding implementation (GNU General Public License v3.0) to sample synthetic gene expression data. The gene expressions are simulated by a stationary dynamical system over a sparse, acyclic regulatory network encoded by $\mathbf{G}$. To simplify the experimental setup, we use the clean gene expressions without technical measurement noise as the observations.

SERGIO models the mRNA concentration of the genes using the chemical Langevin equation, a nonlinear geometric Brownian motion model driven by two independent Wiener processes for each gene. The expression x_j of gene j is primarily defined through its production rate p_j, which depends nonlinearly on the expression levels $\mathbf{x}$ of the other genes through the signed interaction parameters $\mathbf{K}$ and the regulatory network $\mathbf{G}$. Following Dibaeinia and Sinha (2020), we use a Hill nonlinearity coefficient of 2 and sample the parameters k_{ij} as well as 10 master regulator rates b_{jc}, which model cell type heterogeneity, from $k_{ij} \sim \mathrm{Unif}(-5,-1) \cup (1,5)$ and

$b_{jc} \sim \mathrm{Unif}(1, 4)$, respectively. Finally, we use an expression decay rate of $\lambda = 0.5$ and noise scale of $q = 0.5$, which deviates from the values 0.8 and 1.0, respectively, used by Dibaeinia and Sinha (2020) when simulating $d \geq 100$ genes. Under their settings, the data of smaller networks does not contain sufficient signal for the any of the benchmarked methods to learn a nontrivial model of the system.

Interventions Given the fully-specified gene regulation model, we sample an observational (wild-type) dataset and interventional data for single-variable gain-of-function (overexpression) interventions (e.g., Norman et al., 2019) on all genes, each with 1000 measured cell observations, as the interventions for the benchmark. We evaluate overexpression rather than knockdown perturbations, because the former are qualitatively more similar to the test-time shift interventions used to query the models learned by the methods. The gain-of-function interventions are implemented by multiplying the production rate p_j of the target gene j by a randomly-sampled factor $r_j \sim \mathrm{Unif}(2, 10)$. The half-response levels for the Hill nonlinearities are kept at the values estimated during the wild-type simulation, so that the intervention effects propagate downstream.

D.2 Metrics

We focus on comparing the true and predicted interventional distributions of unseen interventions in a system. This evaluation setting mimics applications, but benchmarking different algorithms requires some care. In general, there is a mismatch between the perturbation implemented by an intervention in the ground-truth system and the *query* perturbation performed in a learned causal model—not only because the true model perturbation is unknown, but also because true and learned models may be from different model classes.

Test-time interventions To compare different models at test-time, we query each learned model by a shift intervention on the target variable that induces the same target variable mean as the true, held-out interventional data (Rothenhäusler et al., 2015; Zhang et al., 2021). After performing the intervention, our metrics compare the predicted and true interventional joint distributions. We perform shift interventions, because they have the same definition in both SCMs (1) and stationary diffusions (8). For both, we add a scalar δ to the mechanism $f_j(\mathbf{x})$ of the target variable j (see Appendix D.1.2). To make the query well-defined, we assume knowledge of the true interventional mean of the target variable.

For acyclic SCMs, the query shift δ is given by the difference between the empirical observational mean of the learned SCM and the target interventional mean. However, cyclic SCMs and stationary diffusions may model feedback on the target variable, where the above does not hold. For these models, we find the query shifts δ by an exponential search around $\delta = 0$ for a range estimate $(\delta_{\mathrm{lo}}, \delta_{\mathrm{hi}})$. For each shift, we simulate data from the intervened model and compare the predicted empirical mean to the target mean of the intervened variable. Given an estimated range $(\delta_{\mathrm{lo}}, \delta_{\mathrm{hi}})$, we run a grid search for the optimal value $\delta \in \{\delta_{\mathrm{lo}}, \delta_{\mathrm{lo}} + 1/10(\delta_{\mathrm{hi}} - \delta_{\mathrm{lo}}), \dots, \delta_{\mathrm{lo}} + 9/10(\delta_{\mathrm{hi}} - \delta_{\mathrm{lo}}), \delta_{\mathrm{hi}}\}$. Ultimately, we select the shift δ achieving the minimum distance to the target interventional mean.

Metrics Both metrics we report are computed based on samples from the interventional distributions, which enables a nonparametric comparison across the different models. For each test intervention, we simulate 1000 samples $\tilde{\mathbf{x}}^{(n)} \in \tilde{D}$ from the interventional distribution of the predicted model and compare them with the true interventional dataset of 1000 samples $\mathbf{x}^{(n)} \in D$.

To evaluate the overall fit of the predicted distribution, we compute the Wasserstein distance W_2 to the ground-truth interventional data. To make W_2 efficiently computable, we report the W_2 distance with small entropic regularization, which interpolates between W_2 and the MMD (Genevay et al., 2018). The entropy-regularized W_2 distance between the empirical measures of the datasets D and $\tilde{D}$ with $|D| = M$ and $|\tilde{D}| = N$ is defined as

$$W_2(D, \tilde{D}) := \left(\min_{\mathbf{P} \in U} \sum_{m=1}^{M} \sum_{n=1}^{N} p_{mn} \|\mathbf{x}^{(m)} - \tilde{\mathbf{x}}^{(n)}\|_2^2 - \epsilon H[\mathbf{P}] \right)^{1/2},$$

where H is the entropy defined as $H[\mathbf{P}] := -\sum_{nm} p_{mn}(\log p_{nm} - 1)$, and U is the set of transport matrices $U = \{\mathbf{P} \in \mathbb{R}_{\geq 0}^{M \times N} : \mathbf{P}\mathbf{1}_N = 1/M\, \mathbf{1}_M \text{ and } \mathbf{P}^\top \mathbf{1}_M = 1/N\, \mathbf{1}_N\}$ with $\mathbf{1}_N$ being a vector of N ones (Peyré et al., 2019). For evaluation, the W_2 metric is more robust than the MMD, because it does not depend on the sensitive choice of a kernel bandwidth (Gretton et al., 2012). For $\epsilon > 0$, $W_2(D, \tilde{D})$ can be efficiently computed using the Sinkhorn algorithm. We use the `ott-jax` package (Apache 2.0 Licence) and $\epsilon = 0.1$ (Cuturi et al., 2022).

In addition to the overall fit, we separately assess the accuracy of the interventional means alone. Following Zhang et al. (2022), we report the mean squared error of the predicted empirical means of the d variables given by

$$\mathrm{MSE}(D, \tilde{D}) := \frac{1}{d}\sum_{j=1}^{d}(m_j - \tilde{m}_j)^2\,,$$

where $\mathbf{m} := \frac{1}{M}\sum_{m=1}^{M}\mathbf{x}^{(m)}$ and $\tilde{\mathbf{m}} := \frac{1}{N}\sum_{n=1}^{N}\tilde{\mathbf{x}}^{(n)}$ are the empirical means of the datasets.

D.3 Hyperparameter Tuning

In the experiments, we benchmark the methods on different generative processes (Appendix D.1). To calibrate the important hyperparameters of the methods, we perform cross-validation prior to the final evaluation that benchmarks the methods. All methods are tuned separately for each data-generating process, that is, for cyclic linear SCMs, cyclic linear SDEs, and the gene expression data, for Erdős-Rényi and scale-free sparsity structures.

The experiments for all generative processes are repeated for 50 randomly-sampled systems. For each system, we generate an observational and 10 interventional datasets for learning a model as well as 10 interventional datasets for evaluation, with all interventions performed on different target variables. To tune the hyperparameters of the methods, we split the 10 observed interventions into 9 training and 1 validation dataset. The methods then infer a causal model based on the 9 training interventional and the observational dataset, and we compute the W_2 metric for the unseen validation intervention. For each method, we select the hyperparameter configuration achieving the lowest median W_2 metric on 20 randomly-selected tasks.

D.4 Stationary Diffusions

Models We evaluate linear and nonlinear stationary diffusion models. Both classes of SDE systems model d independent drift and diffusion mechanisms f_j and σ_j that are defined by separate parameters $\boldsymbol{\theta}_j$. For both models, the corresponding group lasso regularizers $R(\boldsymbol{\theta}_j)$ penalize the dependence on the other variables. The models and regularizers are defined as

$$\begin{aligned} f_{\boldsymbol{\theta}_j}(\mathbf{x})_j &= b^j + \mathbf{w}^j \cdot \mathbf{x} & R(\boldsymbol{\theta}_j) &= \textstyle\sum_{i\neq j}^{d}|w_i^j| \\ f_{\boldsymbol{\theta}_j}(\mathbf{x})_j &= b^j + \mathbf{w}^j \cdot g(\mathbf{U}^j\mathbf{x} + \mathbf{v}^j) - x_j & R(\boldsymbol{\theta}_j) &= \textstyle\sum_{i\neq j}^{d}\|\mathbf{u}_i^j\|_2 \end{aligned}$$

where $g(z) := \exp(z)/(\exp(z)+1)$ the sigmoid nonlinearity, applied elementwise. The diffusion term σ is modeled as a constant diagonal matrix $\sigma(\mathbf{x}) = \mathrm{diag}(\exp(\log\boldsymbol{\sigma}))$, with $\log\boldsymbol{\sigma} \in \mathbb{R}^d$. The parameters $\log\boldsymbol{\sigma}$ are learned in log-space to enable gradient-based optimization. To remove the speed scaling invariance, we fix $w_j^j = -1$ in the linear and $\mathbf{u}_j^j = \mathbf{0}$ in the MLP model (see Section 4.4). In the experiments, the MLP model uses a hidden size of $h = 8$ for the matrices $\mathbf{U}^j \in \mathbb{R}^{h\times d}$ and vectors $\mathbf{v}^j, \mathbf{w}^j \in \mathbb{R}^h$.

Interventions during training For both SDE models, we model the drift mechanisms $f_{\boldsymbol{\theta},\boldsymbol{\phi}}(\mathbf{x})_j$ of the variables targeted by interventions as shift interventions as defined in (8, left) with parameters $\boldsymbol{\phi}_j = \{\delta_j\}$. As described in Algorithm 1, we learn the parameters $\boldsymbol{\phi}_j$ jointly with $\boldsymbol{\theta}$, since they are unknown. For the purpose of the experiments, we limit the learned interventions to shifts in order to allow a direct comparison with SCMs. However, we found learning more complex intervention parameterizations like, for example, full shift-scale interventions as in (8), generally straightforward. More expressive interventions shift some of the burden of explaining the distribution shift from $\boldsymbol{\theta}$ to $\boldsymbol{\phi}_j$, which can help inferring robust parameters $\boldsymbol{\theta}$ under model mismatch.

Optimization For all experiments, we run 20,000 update steps on the KDS as described in Section 4.4 using the Adam optimizer with learning rate 0.001. We compute the empirical KDS (5) using the Gaussian kernel k_γ and a batch size of $|D| = 512$. For initialization of the parameters $\boldsymbol{\theta}$, we use a zero-mean Gaussian for the linear model and LeCun-Uniform initialization for the nonlinear model, respectively, both with scale 0.001. We initialize the intervention shifts $\boldsymbol{\phi}_j = \{\delta_j\}$ by warm-starting them at the difference in means of the target variable in the interventional and the observational datasets. Overall, the important hyperparameters are the kernel bandwidth γ and the group lasso regularization strength λ, so we tune these for each experimental setting using the protocol described in Appendix D.3.

Table 1: **Hyperparameter tuning for the experiments in Section 6.** The hyperparameters of all methods are selected using the protocol described in Appendix D.3.

Method	Hyperparameter	Search range
IGSP	significance level	$\alpha_{\text{IGSP}} \in \{0.001, 0.003, 0.01, 0.03, 0.1\}$
DCDI	sparsity regularization	$\lambda_{\text{DCDI}} \in \{0.001, 0.01, 0.1, 1, 10\}$
	number of MLP layers	$m_{\text{DCDI}} \in \{1, 2\}$
NODAGS	sparsity regularization	$\lambda_{\text{NODAGS}} \in \{0.0001, 0.001, 0.01, 0.1\}$
	spectral norm terms	$n_{\text{NODAGS}} \in \{5, 10, 15\}$
	learning rate	$\eta_{\text{NODAGS}} \in \{0.001, 0.01, 0.1\}$
	hidden units	$m_{\text{NODAGS}} \in \{1, 2, 3\}$
LLC	sparsity regularization	$\lambda_{\text{LLC}} \in \{0.001, 0.01, 0.1, 1, 10, 100\}$
KDS	sparsity regularization	$\lambda \in \{0.001, 0.003, 0.01, 0.03, 0.1\}$
	kernel bandwidth	$\gamma \in \{3, 5, 7\}$

Diagnostics The following intuitions may be helpful when deploying our inference approach. If the stationary density induced by the learned SDEs overfits or collapses to a small part of the data, the kernel bandwidth may be too small. Our bandwidth range is suitable for standardized datasets of $d = 20$ variables but should likely be expanded in different settings. If the learned SDEs are unstable upon convergence or diverge during simulations—despite a decreasing or near-zero KDS loss—then the speed scaling invariance may not be adequately fixed (see above and Section 4.4). In this context, we find that the fit and performance of the models empirically improves when fixing the self-regulating parameters of f_j on x_j, rather than, e.g., the noise scales σ_j. Without any sparsity regularization, Algorithm 1 may converge to models at the edge of stability, e.g., to linear models with maximum real parts of the eigenvalues being near zero and only just negative. Sparsity regularization can mitigate such instability and related issues.

D.5 Baselines

GIES (Hauser and Bühlmann, 2012) assumes a linear-Gaussian SCM to infer a graph equivalence class, from which we randomly sample a causal graph. To perform the greedy search, we run the original `R` implementation of the authors using the Causal Discovery Toolbox (MIT Licence)[4]. Given the DAG estimate, we use a linear-Gaussian SCM with maximum likelihood parameter and variance estimates as the learned model. These estimates have simple closed-forms that account for interventional data (Hauser and Bühlmann, 2012). At test time, the shift interventions are implemented in the learned linear SCM and the data sampled as described in Appendix D.1.2.

IGSP (Wang et al., 2017) uses a Gaussian partial correlation test. We use the same closed-form maximum likelihood parameter and variances estimates as for GIES to construct the final causal model. For IGSP, we run the implementation provided as part of the CausalDAG package (3-Clause BSD License)[5]. Using the protocol described in Appendix D.3, we tune the significance level α_{IGSP} of the conditional independence test for each experimental setting individually by searching over a range of α_{IGSP} values (see Table 1). As for GIES, the shift interventions are implemented in the estimated linear SCM as described in Appendix D.1.2.

DCDI (Brouillard et al., 2020) learns a nonlinear, Gaussian SCM parameterized by neural networks jointly with the noise variance. For comparison with the nonlinear stationary diffusion model, we use the same hidden size of 8 for the neural networks. To run DCDI, we use the Python implementations provided by the authors (MIT License). We tune the regularization strength λ_{DCDI} and the number of layers m_{DCDI} and leave the remaining hyperparameters at the suggestions by the authors (see Table 1). When learning from imperfect interventions, DCDI estimates a separate model for each interventional environment. For evaluation, we use the model learned for the observational dataset and implement the shift interventions by adding the bias δ to the mean of the Gaussian modeling the target variable, analogous to the linear SCMs and SDEs described in Appendix D.1.2.

[4] https://github.com/FenTechSolutions/CausalDiscoveryToolbox
[5] https://github.com/uhlerlab/causaldag

NODAGS (Sethuraman et al., 2023) infers a nonlinear cyclic SCM using residual normalizing flows and also estimates the noise variances. As suggested by the authors, we jointly tune the regularization parameter λ_{NODAGS}, the number of terms for computing the spectral norm n_{NODAGS}, the learning rate η_{NODAGS}, and the number of hidden units m_{NODAGS} (see Table 1). We set the remaining hyperparameters to the recommendations by the authors and use the implementation published alongside the original paper (Apache 2.0 Licence). At test time, the shift interventions are implemented in the model by using $\mathbf{U} = \mathbf{I}$ and otherwise as described in the paper, analogous to the linear cyclic SCMs described in Appendix D.1.2 (see also Hyttinen et al., 2012).

LLC (Hyttinen et al., 2012) learns a linear cyclic SCM and estimates the noise variances. For the basic implementation of the LLC algorithm, we use the code provided by the NODAGS repository. However, we extend their implementation by the ℓ_1 sparsity regularizer described in Section 6.2 of the original paper by Hyttinen et al. (2012), solving the minimization problem with BFGS. We treat the weight λ_{LLC} of this regularizer as a hyperparameter that is tuned via a grid search (see Table 1). At evaluation time, the shift interventions in the learned cyclic linear SCM are performed as for GIES and IGSP.

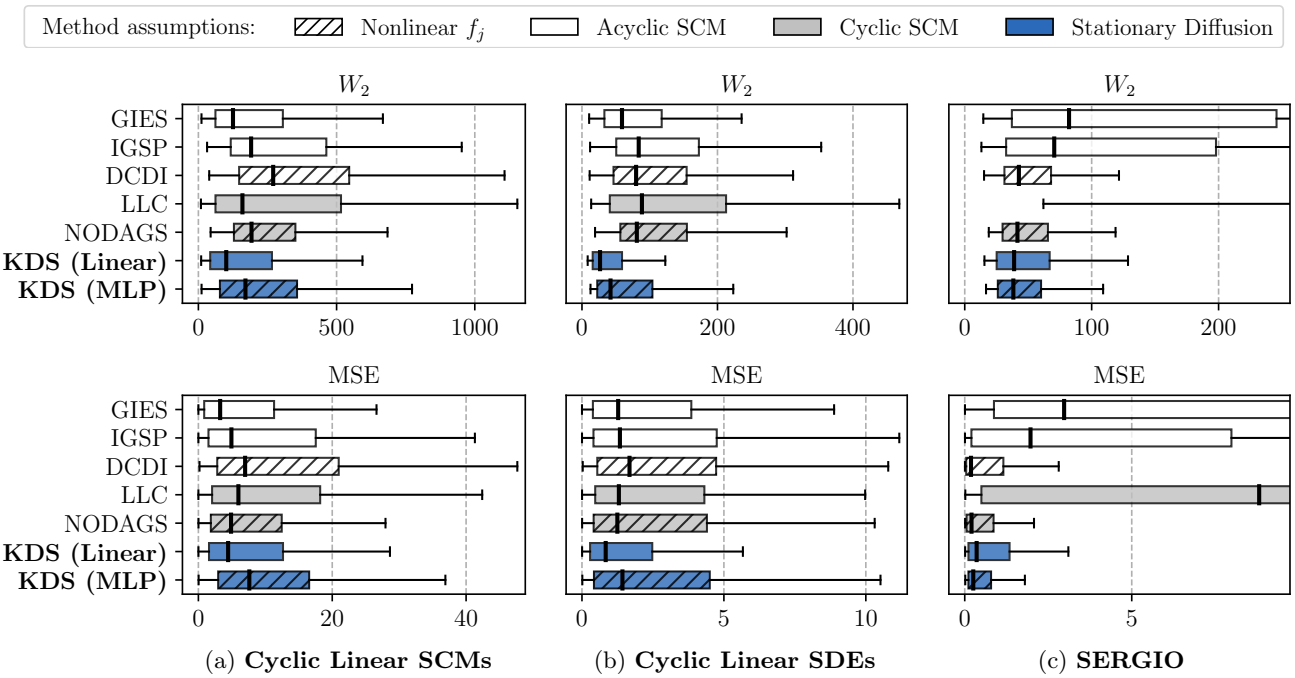
D.6 Compute Infrastructure

The development and experiments of this work were carried out on an internal cluster. In each experiment, all methods ran for up to 1 hour of wall time on up to 4 CPUs and 16 GB of RAM, adjusted individually according to the compute requirements of each method. We implement our approach with `JAX` (Bradbury et al., 2018) and thus additionally provide 1 GPU, which allows for significant speed-ups during development and the experiments. Overall, running our inference method takes approximately one hour given the above resources, both for the linear and nonlinear model, and including the final search for test-time intervention shifts.

E Additional Results

Figure 6 presents supplementary experimental results. The setup is the same as in Figure 4, except that the causal dependency structure of the ground-truth systems is scale-free rather than Erdős-Rényi (see Appendix D.1.1). Similar to Figure 4, the stationary diffusion models, in particular those with linear mechanisms $f_{\boldsymbol{\theta}}$, are the most accurate at predicting the effects of the unseen interventions overall. The MLP variants achieve slightly worse results compared to Figure 4 but are still competitive with the nonlinear SCM baselines. Here, stationary diffusions may further improve if the sparsity regularizer described in Appendix D.4 groups out- rather than ingoing dependency parameters, inducing power law out- rather than in-degree degree structure.

Since some of the reported metrics exhibit high variance, we additionally ran Wilcoxon signed-rank tests (one-sided, significance level $\alpha = 0.05$). The results are shown in Table 2. We find that our approach significantly outperforms the baselines across most metrics and data settings.



(a) **Cyclic Linear SCMs** (b) **Cyclic Linear SDEs** (c) **SERGIO**

Figure 6: **Benchmarking results** ($d = 20$ variables, scale-free causal structure). Metrics are computed from 10 test interventions on unseen target variables in 50 randomly-generated systems. Box plots show medians and interquartile ranges (IQR). Whiskers extend to the largest value inside 1.5 times the IQR length from the boxes.

Table 2: **Significance tests supporting the experimental results.** The table shows $(*/*)$ if the alternative hypothesis—our approach outperforming the baselines on average, i.e., the metric distribution being in our favor—was accepted for either metric (W_2/MSE) using a Wilcoxon signed-rank tests (one-sided, significance level $\alpha = 0.05$).

	KDS (Linear)			**KDS (MLP)**		
Figure	4a	4b	4c	4a	4b	4c
GIES	*/*	*/*	*/*	*/	*/*	*/*
IGSP	*/*	*/*	*/*	*/*	*/*	*/*
DCDI	*/*	*/*	*/	*/*	*/*	*/
LLC	*/*	*/*	*/*	*/*	*/*	*/*
NODAGS	*/*	*/*	*/	*/*	*/*	*/

	KDS (Linear)			**KDS (MLP)**		
Figure	6a	6b	6c	6a	6b	6c
GIES	*/	*/*	*/*	/	*/	*/*
IGSP	*/*	*/*	*/*	*/	*/	*/*
DCDI	*/*	*/*	*/	*/*	*/	*/
LLC	*/*	*/*	*/*	*/	*/	*/*
NODAGS	*/*	*/*	*/	*/	*/	*/