Bounds on Extrapolation across Phase Transitions with Generalized Regression

Jeffrey Wei [1] **Manolis Zampetakis** [1] **John Sous** [1]

Abstract

Phase transitions occur when a physical system undergoes a dramatic transformation as it crosses a transition. We study whether regression methods trained only in one phase can predict the physical properties that characterize the unobserved phase. Our test sets consist of the analytically tractable Heisenberg model and angle-resolved photoemission spectroscopy measurements of a venerable high transition temperature superconductor. We find that polynomial regression and kernel methods recover the unobserved phase structure, while standard neural networks fail. To bound the error, we adopt the generalized eigenvalue problem (GEVP) into an upper bound on the worst-case transfer coefficient, defined as the ratio of mean squared error (MSE) loss on the extrapolation region to the MSE loss on the training region. Empirically, we find that our GEVP bound stays approximately constant with respect to the empirical transfer coefficient across training-set sizes and distance between the training and extrapolation region for both datasets, providing a principled framework for studying extrapolation feasibility.

1. Introduction

A major goal of theoretical physics is to make predictions in regimes where information is limited or inaccessible. This objective can be reframed as an extrapolation problem: can we infer behavior in regimes where the physics is intractable from regimes where it is tractable? Continuous phase transitions are a canonical example, where a smooth underlying physical observable can give rise to nonanalytic behavior in derived quantities such as an order parameter at a transition point. Understanding phase transitions is important across science, spanning the behavior of matter in the early universe, the properties of magnets and superconductors, everyday changes such as water freezing or ice melting, and phase structure in learning theory.

Vargas-Hernández et al. (2018) empirically showed that Gaussian processes with iteratively constructed kernels (Duvenaud et al., 2013) can extrapolate across phase transitions. However, their approach evaluates extrapolation quality only after comparing to ground truth in the extrapolation region. It does not provide a principled error bound for the prediction before target-phase data is available. This limits its usefulness in settings where the target phase is experimentally inaccessible or theoretically unresolved.

In this work, we develop bounded extrapolation methods for making reliable theoretical physics predictions across phase transitions. We train polynomial regression, kernel methods — Gaussian processes and random Fourier features (RFF) — and neural network regression on data from an observed phase and ask whether they can extrapolate the physical observable of the system into an unobserved phase (Rasmussen & Williams, 2006; Rahimi & Recht, 2007). We evaluate this question in two complementary settings: a mean-field Heisenberg model of a ferromagnetic transition, where the underlying physical theory is analytically known, and experimentally obtained angle-resolved photoemission spectroscopy (ARPES) data, where the observable is noisy and measured from a real high-transition-temperature superconductor (Landau & Lifshitz, 1980; He et al., 2021). Across both settings, polynomial regression, RFF, and Gaussian processes extrapolate the physical observable across the phase boundary, while neural networks, including vanilla MLPs, Fourier-feature MLPs (Tancik et al., 2020), and SIREN (Sitzmann et al., 2020), do not.

To quantify extrapolation reliability, we characterize the transfer coefficient, which measures how much the mean-squared prediction error for the physical observable increases from the training region to the extrapolation region. For learned models with a finite number of features, such as degree-d polynomial features or m random Fourier features, this transfer coefficient can be computed by solving a generalized eigenvalue problem (GEVP), which we call our GEVP bound. Empirically, the GEVP bound remains nearly constant relative to the observed transfer coefficient as we vary the number of training samples.

[1]Yale University, New Haven, Connecticut, United States. Correspondence to: Jeffrey Wei <jeffrey.wei@yale.edu>.

Proceedings of the AI4Physics Workshop at the 43rd International Conference on Machine Learning (AI4Physics@ICML 2026), Seoul, South Korea. 2026. Copyright 2026 by the author(s).

A key empirical finding is that extrapolation difficulty has structure. As we move the training region farther from the transition, the empirical transfer coefficient grows approximately as a power law in the distance to the transition. The GEVP bound follows the same distance-dependent power law, demonstrating sensitivity to the geometry of the extrapolation problem rather than acting as a fixed loose constant. This provides evidence that extrapolation via polynomial and kernel regression across phase boundaries may be feasible and principled.

Our main contributions are as follows.

1. We formalize phase-transition extrapolation as a class of learning problems where the target behavior in the extrapolation regime is dramatically different from that observed during training. Because the change is sharp, structured, and physically interpretable, this setting provides a clear benchmark for studying principled extrapolation beyond the training regime.
2. We show that polynomial regression and kernel methods extrapolate more reliably than the neural network methods we test on phase transitions.
3. We adapt disjoint-support transfer theory, extremal-polynomial bounds, and finite-feature spectral methods to bound the empirical transfer coefficient. The GEVP bound is our tightest bound, staying approximately constant relative to the observed transfer coefficient for polynomials and kernels across both datasets. We also find an approximate power-law relationship between extrapolation distance and transfer error, demonstrating that the empirical transfer coefficient and the GEVP bound both grow in a controlled way.

Related Work. Several machine-learning papers use supervised and unsupervised models to detect, classify, or locate phases from simulation or experimental data (Carrasquilla & Melko, 2017; van Nieuwenburg et al., 2017; Wang, 2016; Wetzel, 2017). In constrast, we ask whether a model trained only in one phase can extrapolate the physical observable into an unobserved phase. Vargas-Hernández et al. (2018) showed that Gaussian processes with iteratively constructed compositional kernels can extrapolate across analytic phase transitions, building on earlier work on automatic kernel construction (Grosse et al., 2012; Duvenaud et al., 2013; Lloyd et al., 2014; Wilson & Adams, 2013). Their results are empirical and require ground truth in the extrapolation region to assess success, while our goal is to provide a bound to extrapolation quality before target-phase data is available.

Gaussian processes are kernel methods that define a distribution over functions through a covariance kernel (Rasmussen & Williams, 2006). Random Fourier features provide a finite-dimensional representation of translation-invariant kernels by approximating them with Monte Carlo sinusoidal features (Rahimi & Recht, 2007). More generally, Bach (2017) studies random feature expansions for positive-definite kernels through their reproducing kernel Hilbert space (RKHS), which is the Hilbert space of functions associated with a kernel. Rudi & Rosasco (2017) gives generalization guarantees for learning with such random features. These results motivate our use of RFFs as a finite-dimensional kernel representation for which the GEVP bound can be computed directly.

Our disjoint-support between train and extrapolation region falls outside much of standard transfer-learning theory. The closest theoretical result to our setting is the non-overlap polynomial transfer bound of Kalavasis et al. (2024). We build on this line by comparing it to a multivariate Remez bound (Brudnyi & Ganzburg, 1973) and to a GEVP bound that uses specific sample geometry of the training and extrapolation regions to tighten the bound.

2. Problem Setup

Disjoint-Support Regression. Our target is a regression on a physical observable function $f^* : \mathcal{X} \to \mathbb{R}$ on a domain $\mathcal{X} = \mathcal{T} \times \mathcal{Y} \subset \mathbb{R}^2$, with $T \in \mathcal{T}$ a tuning parameter and the second coordinate system-specific order parameter, where the phase transition occurs. The domain is partitioned into two disjoint rectangles: a training region P entirely above T_c, where the model is fit, and an extrapolation region Q that extends across T_c and contains the target phase on which the model is evaluated. We denote by D the distance from the lower-temperature edge of P to the transition point T_c, so increasing D moves the observed training region farther from the transition while leaving the extrapolation task fixed.

We take P and Q to be uniform measures over their respective domains. At training time, we uniformly draw N samples from the training region $\{(x_i, y_i)\}_{i=1}^N$ with $x_i \sim P$ and $y_i = f^*(x_i) + \eta_i$ and fit a learner $\hat{f}$ by empirical risk minimization on these samples. At test time, we report the validation error $\mathrm{MSE}_P(\hat{f}) = \mathbb{E}_{x\sim P}[(f^*(x) - \hat{f}(x))^2]$ on the 20% validation split and extrapolation error $\mathrm{MSE}_Q(\hat{f}) = \mathbb{E}_{x\sim Q}[(f^*(x) - \hat{f}(x))^2]$. Hyperparameters are selected on an 80/20 train and validation split inside P alone, forcing our learner $\hat{f}$ to strictly learn the physics of the training region.

For any function class $\mathcal{F}$ that contains the residual $r = f^* - \hat{f}$, the worst-case transfer coefficient $\Gamma(\mathcal{F}; P, Q) := \sup_{g\in\mathcal{F},\, g\neq 0} \mathbb{E}_{x\sim Q}[g(x)^2]/\mathbb{E}_{x\sim P}[g(x)^2]$ depends only on the geometry of P, Q, and the chosen class, which yields $\mathrm{MSE}_Q(\hat{f}) \leq \Gamma(\mathcal{F}; P, Q) \cdot \mathrm{MSE}_P(\hat{f})$. This quantity converts a validation MSE in the training region into an extrapolation MSE, and every bound we state below is just a particular choice of $\mathcal{F}$ followed by a computation of Γ.

The residual formulation is useful only if $\mathcal{F}$ is expressive enough to contain $r = f^* - \hat{f}$. The function classes we consider — polynomials, random Fourier features, Gaussian processes, neural networks — are all universal approximators in the limit of unbounded degree, RKHS norm, or width, so the residual in principle is captured by any of our chosen regression methods. However, that same universality is exactly why Γ can explode. There is a fundamental tradeoff between expressiveness and the ability to extrapolate: a larger $\mathcal{F}$ fits f^* better on P but admits more functions consistent with the training data, each of which can disagree wildly on Q. Put simply, without some restriction on $\mathcal{F}$, extrapolation across a disjoint domain is impossible. Thus, an important focus of our paper is to explore a small enough $\mathcal{F}$ that our bounds are finite and informative while keeping it large enough to approximate the physical observable.

In our setting, this intuition is supplied by Landau theory (Landau & Lifshitz, 1980), which models the free energy near a continuous phase transition as a low-order polynomial expansion in the order parameter, with coefficients that vary smoothly with the tuning parameter. This directly motivates a low-degree polynomial hypothesis class for the analytic Heisenberg model. For the ARPES measurements, we assume that away from the critical point, a smooth measured function of temperature and energy can be locally approximated by low-order polynomial terms. This motivates using low-order polynomials as a controlled empirical hypothesis class for the measured ARPES intensity.

Model mismatch is mainly a concern for ARPES. In the Heisenberg case, the polynomial bias is matched to the Landau free energy, while in ARPES the low-degree polynomial is only an empirical approximation to the measured intensity. Nevertheless, we find strong extrapolation results on ARPES as well, validating our intuition that a restricted feature class can still capture the dominant structure of the measured observable.

Phase Transitions. Both physical systems are continuous phase transitions. Successful extrapolation should recover not only the physical observable structure in the target region, but also the phase transition structure extracted from the surface. For Heisenberg, this is the magnetization obtained by minimizing the predicted free energy, and for ARPES, this is the migration of the spectral peak associated with the superconducting gap. We pair an analytic target with an experimental target so that the same protocol is tested both where the underlying physics is known in closed form and where the observable is measured from real data. The dataset preprocessing details are deferred to Appendix C. Both targets are visualized in Figure 2.

Heisenberg (analytic). For the analytic target, we predict the free-energy surface $f(T, m)$ as a function of temperature T and magnetization m. The mean-field Landau–Ginzburg free energy of a ferromagnetic Heisenberg model retains the leading m^2 and m^4 terms (Landau & Lifshitz, 1980):

$$f(T, m) = \tfrac{1}{2}(1 - T_c/T)\, m^2 + \tfrac{1}{12}(T_c/T)^3\, m^4 + c_0. \quad (1)$$

For $T > T_c$, the free-energy minimum occurs at $m = 0$, corresponding to the paramagnetic phase. For $T < T_c$, the surface develops two symmetric minima at nonzero magnetization, corresponding to the ferromagnetic phase. The order parameter is the magnitude of the minimizing magnetization of the free energy, $|m^*(T)| = |\arg\min_m f(T, m)|$.

ARPES for high-T_c cuprate (experimental). For the experimental target, we predict the ARPES intensity $I(T, E)$ as a function of temperature T and binding energy E. ARPES measures the electronic spectral function in the Pb-doped Bi-2212 cuprate of He et al. (2021). For ARPES, the measured intensity is a physical observable that responds to the superconducting transition, but it is not itself the thermodynamic quantity that defines the transition. We therefore use the experimentally reported T_c to define the training and extrapolation regions, and we treat the recovered peak energy $E^*(T)$ as a visible indicator of the superconducting gap rather than as a mathematical way to calculate T_c. Unlike the Heisenberg target, the ARPES intensity has no closed-form expression known to us.

3. Bounds on Extrapolation

We compare four learners fit on samples from the training region P and evaluated on the extrapolation region Q. Per-method training details and hyperparameter grids are deferred to Appendix E.

MLPs. Multi-Layer Perceptrons (MLPs) are universal approximators that fit arbitrary smooth targets effortlessly inside the training distribution. The MLP fits f^* on P to numerical precision, but the failure mode on Q is structural: Xu et al. (2021) prove that ReLU MLPs extrapolate linearly outside the training convex hull, and similar poor extrapolation results are demonstrated empirically on our chosen activation functions GELU and tanh. We also benchmark a Fourier-feature MLP (FF-MLP) (Tancik et al., 2020), Fourier feature input embeddings motivated by the success of RFFs, and SIREN (Sitzmann et al., 2020), sinusoidal activations in every layer, as the neural network methods engineered for high-frequency targets. We show that all chosen neural network methods fail to recover the unobserved phase structure in Table 2 and Figure 6 of Appendix D. For neural networks, we do not have a systematic way to bound its transfer coefficient. This aligns with their empirically unstable extrapolation behavior in our tests.

Polynomials. We fit $\hat{f}$ using ridge regression on a multivariate polynomial feature expansion of the input coor-

dinates. The polynomial degree d controls the maximum total degree of the expansion and is selected using the 80/20 holdout inside P. For Heisenberg, the input coordinates are (T, m), or $(1/T, m)$ when using the reciprocal-temperature basis. For ARPES, the input coordinates are (T, E).

Three bounds for the polynomial residual class $\mathbb{P}_d$ capture increasingly problem-specific ways to control how a polynomial residual $r = f^* - \hat{f}$ can grow from P to Q.

Proposition 3.1 (Volume bound, Kalavasis et al., 2024)**.** *Let $P, Q \subset R_{\text{full}} \subset \mathbb{R}^n$ be disjoint rectangles with uniform measure, and define $\rho_P = |R_{\text{full}}|/|P|$ and $\rho_Q = |R_{\text{full}}|/|Q|$. If $r = f^* - \hat{f} \in \mathbb{P}_d$, then*

$$\mathrm{MSE}_Q(r) \leq (2d)^{2d}\, \rho_Q\, \rho_P^{2d}\, \mathrm{MSE}_P(r).$$

The volume bound is geometry-based and applies under disjoint support without using samples or labels in Q. This bound is found to be very loose for our problem setup as shown in Figure 4. The reason is that the bound is agnostic over a broad family of possible pairs P, Q with the same relative volumes, so it cannot exploit the specific geometry and uniformity of our training and extrapolation rectangles. Because the bound must hold for many possible configurations with the same relative volumes, the exponential dependence on d is unavoidable in the worst case.

Proposition 3.2 (Remez bound, Brudnyi & Ganzburg, 1973)**.** *Let $R_{\text{full}} \subset \mathbb{R}^n$ be convex, let $P \subset R_{full}$ have relative volume $\lambda = |P|/|R_{full}|$, and let $r \in \mathbb{P}_d$ satisfy $\varepsilon = \sup_{x \in P} |r(x)|$. Then*

$$\mathrm{MSE}_Q(r) \leq \left(\frac{4n}{\lambda}\right)^{2d} \varepsilon^2.$$

The Remez bound provides a different geometry-based bound. It replaces the training MSE with the maximum residual error on P, and uses the multivariate Remez inequality to bound how large the same polynomial can become on the enclosing region. This creates a tradeoff: the dependence on dimension can be favorable in low-dimensional settings like ours ($n = 2$ for T, m on Heisenberg or T, E on ARPES), but the bound can become large when the maximum training error is much larger than the average training error. Similar to the volume bound, it does not exploit the specific geometry of the training and extrapolation rectangles, so it remains exponentially large in the polynomial degree d and can sit far above the empirical transfer coefficient as demonstrated in Figure 4.

Proposition 3.3 (Polynomial GEVP bound)**.** *Let P and Q denote the uniform measures over the training and extrapolation rectangles. Let $\phi_d(x) = (x^\alpha)_{|\alpha| \leq d}$ be the degree-d monomial feature map, and define $A_d = \mathbb{E}_{x \sim Q}[\phi_d(x)\phi_d(x)^\top]$ and $B_d = \mathbb{E}_{x \sim P}[\phi_d(x)\phi_d(x)^\top]$. If B_d is positive definite on the feature span, then*

$$\Gamma(\mathbb{P}_d; P, Q) = \sup_{r \in \mathbb{P}_d,\, r \neq 0} \frac{\mathbb{E}_{x \sim Q}[r(x)^2]}{\mathbb{E}_{x \sim P}[r(x)^2]} = \lambda_{\max}(A_d, B_d).$$

Then, if $r = f^* - \hat{f}$ lies in $\mathbb{P}_d$, then $\mathrm{MSE}_Q(\hat{f}) \leq \lambda_{\max}(A_d, B_d)\, \mathrm{MSE}_P(\hat{f})$.

Because P and Q are known sampling domains, the expectations defining A_d and B_d do not require labeled data from either region. In practice, we estimate them with a large number of unlabeled Monte Carlo samples from the known rectangles. Thus, for a fixed feature map, the GEVP bound depends only on the geometry of P, Q and the feature representation, not on the number of labeled training samples used to fit $\hat{f}$.

The GEVP bound is still a worst-case bound over the degree-d polynomial residual class, but unlike the volume and Remez bounds, it is not worst-case over a broad family of possible pairs (P, Q). Instead, it uses samples from the specific training and extrapolation domains to estimate the feature covariance matrices, so the resulting worst-case direction is adapted to the actual extrapolation problem. This makes the GEVP bound substantially tighter than the volume and Remez bounds in our experiments. Because it is the only bound that stays near the empirical transfer coefficient, we focus our experiments on testing its robustness.

Derivations of all three bounds are given in Appendix A.

Gaussian Processes. Gaussian processes are the most expressive kernel baseline for this problem. Vargas-Hernández et al. (2018) found that Gaussian processes with iteratively built compositional kernels (Duvenaud et al., 2013) extrapolate well across phase transitions, including the Heisenberg free energy that our work expands on and several polaron models (Marchand et al., 2010; Lau et al., 2007; Herrera et al., 2013; Gerlach & Löwen, 1991). We replicate their protocol by searching over compositions of base kernels, fitting each candidate by maximum log marginal likelihood, and selecting kernels by Bayesian information criterion (BIC), a model selection metric to balance model fit with low complexity models (Schwarz, 1978). The details of GP training are deferred to Appendix E.

We wish to reformulate the GEVP bound to Gaussian processes, but the bound is directly computable only for finite feature representations. A Gaussian process is defined by a kernel and its associated reproducing kernel Hilbert space (RKHS), which is generally infinite-dimensional, so it does not provide a finite feature covariance matrix for the same GEVP computation (Aronszajn, 1950; Schölkopf & Smola, 2002; Rasmussen & Williams, 2006). Random Fourier features provide a tractable finite-dimensional approximation for translation-invariant kernels. By Bochner's theorem,

such kernels can be represented as Fourier transforms of spectral measures, and RFFs approximate this representation by Monte Carlo sampling random frequencies from the corresponding spectral measure (Rahimi & Recht, 2007). This gives a finite random feature map on which the GEVP transfer coefficient can be computed.

Random Fourier Features. Random Fourier features approximate a kernel by replacing it with a finite set of random sinusoidal features (Rahimi & Recht, 2007). In our implementation, we draw m features from a Gaussian spectral measure, $z_j(x) = \sqrt{2/m}\cos(\omega_j^\top x + b_j)$, $j = 1, \ldots, m$, with b_j uniform on $[0, 2\pi)$, and fit $\hat{f}$ by ridge regression in the resulting m-dimensional feature space. Once the model has a finite feature map, the same GEVP argument used for polynomials applies exactly.

Proposition 3.4 (RFF GEVP bound)**.** *Let $z : \mathcal{X} \to \mathbb{R}^m$ be the random Fourier feature map, and let $\mathcal{G}_m = \{g_w(x) = w^\top z(x) : w \in \mathbb{R}^m\}$. Let P and Q be the uniform measures over the training and extrapolation domains, and define $A_m = \mathbb{E}_{x\sim Q}[z(x)z(x)^\top]$, $B_m = \mathbb{E}_{x\sim P}[z(x)z(x)^\top]$. If B_m is positive definite on the feature span, then*

$$\Gamma(\mathcal{G}_m; P, Q) = \sup_{g\in\mathcal{G}_m,\, g\neq 0} \frac{\mathbb{E}_{x\sim Q}[g(x)^2]}{\mathbb{E}_{x\sim P}[g(x)^2]} = \lambda_{\max}(A_m, B_m).$$

Then, if $r = f^* - \hat{f}$ lies in $\mathcal{G}_m$, then $\mathrm{MSE}_Q(\hat{f}) \leq \lambda_{\max}(A_m, B_m)\,\mathrm{MSE}_P(\hat{f})$.

Random-feature approximation results explain why RFFs are a natural finite-dimensional proxy for kernel methods (Rahimi & Recht, 2007; Sutherland & Schneider, 2015; Bach, 2017; Rudi & Rosasco, 2017). When the random features are sampled from the spectral measure of the same kernel, and when the kernel, sampling distribution, and number of features satisfy the required assumptions, the finite random-feature space approximates the corresponding kernel RKHS in $L^2(\rho)$ and can achieve comparable statistical guarantees (Bach, 2017; Rudi & Rosasco, 2017). This motivates comparing the RFF empirical transfer coefficient to the GP empirical transfer coefficient.

Our RFF features are sampled from a Gaussian spectral measure, while the GP uses an iteratively built compositional kernel. Therefore, we treat the RFF GEVP as an exact bound for the finite RFF model and an empirical proxy for the GP. In Figure 1, we test this relationship directly: as the number of random features m increases, the RFF empirical transfer coefficient approaches the GP empirical transfer coefficient, while the RFF GEVP upper bounds both. The formal background on random-feature approximation is deferred to Appendix B.

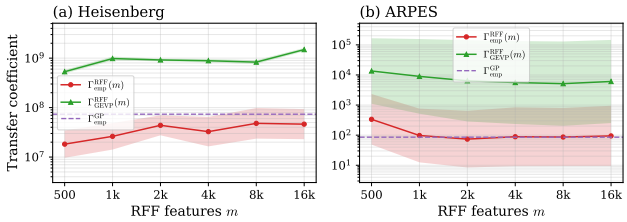


Figure 1. RFF empirical and GEVP transfer coefficients versus number of random features m at fixed $N = 2000$, shown alongside the GP empirical transfer coefficient. Curves show averages across 5 seeds and shaded regions show one standard deviation.

4. Generalized Regression Extrapolates Across Phase Transitions

1: Generalized Regression Empirically Extrapolates

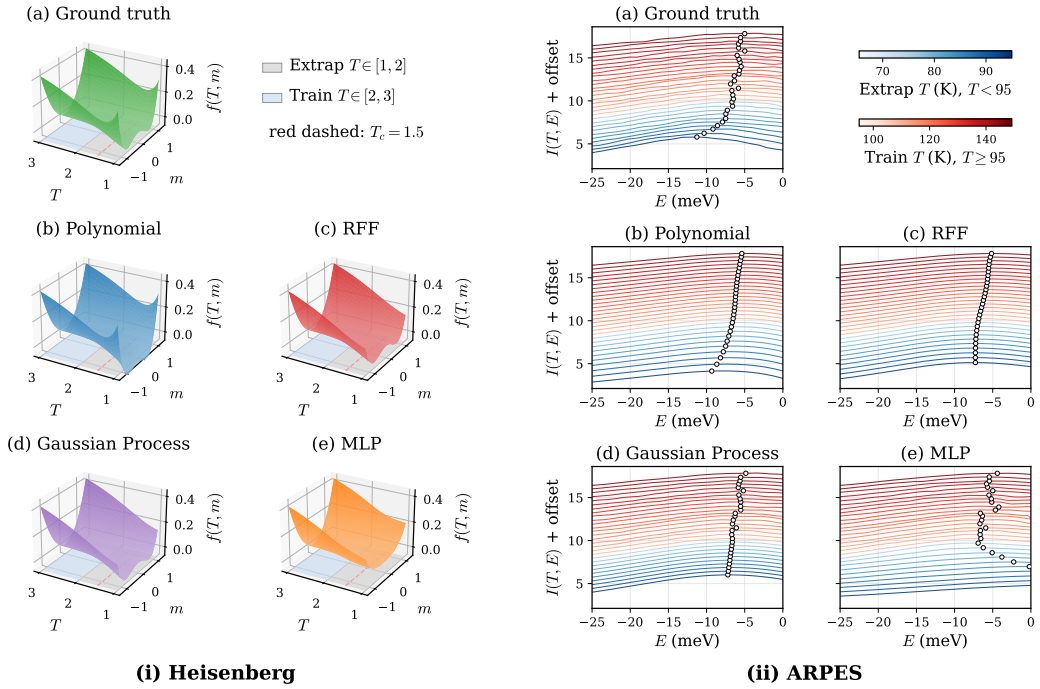
Polynomial regression, RFF regression, and Gaussian processes recover the physical structure across the phase transition on both the Heisenberg model and the experimental ARPES target. The neural-network methods do not recover the same unobserved phase structure.

For Heisenberg, MLPs fail to recover the below-T_c double-well geometry, unlike the polynomial and kernel methods. For ARPES, the MLP failure mode appears as the loss of the temperature-dependent peak migration (as denoted by the marker) across the transition. These failures are clearer in the derived quantities tracking the transition (order parameter for the Heisenberg model and peak ARPES energy for the cuprate) curves in Figure 3, where the MLP misses both the magnetization transition and the ARPES peak-energy migration. This highlights a central extrapolation failure of neural networks, where they can fit the observed phase, but without the right low-dimensional physical bias, they do not infer the new qualitative structure that appears across the phase boundary. Notably, the polynomial and kernel methods predict the Heisenberg critical temperature to approximately 1% accuracy, providing evidence of precise extrapolation.[1] Additional fixed-temperature cuts are provided in Figure 7, and Fourier-feature MLP and SIREN failures are reported in Appendix D.

2: GEVP Tightly Bounds the Extrapolation Error

Across training-set sizes, the GEVP bound robustly upper bounds the observed transfer coefficient, while geometry-based polynomial bounds are much looser. These results go beyond the setting considered by

[1] We omit this metric for ARPES because the measured spectral intensity peak changes across the superconducting transition but does not by itself define T_c.



(i) Heisenberg **(ii) ARPES**

Figure 2. Four-method landscape comparison on the extrapolation region Q for Heisenberg $f(T, m)$ and ARPES $I(T, E)$. Panel (a) in each row shows the ground truth. For ARPES, the marker tracks the recovered peak energy at each temperature.

Table 1. Extrap Rel. % is the extrapolation relative error, defined as the mean absolute error on Q normalized by the standard deviation of the ground truth on Q. $|\Delta T_c|/T_c$ % is the normalized error in predicted T_c expressed as a percentage. Values are averaged across 5 seeds at $N = 2000$.

	Heisenberg			**ARPES**	
Method	Extrap NMSE	Extrap Rel. %	$\lvert\Delta T_c\rvert/T_c$ (%)	Extrap NMSE	Extrap Rel. %
Polynomial	5.1×10^{-2}	7.3	0.13	8.1×10^{-2}	17.8
RFF	2.8×10^{-2}	6.5	0.53	1.8×10^{-1}	22.2
GP	3.3×10^{-2}	5.5	1.13	8.3×10^{-3}	5.4
MLP	2.0×10^{-1}	28.8	—	1.1×10^{-1}	23.1

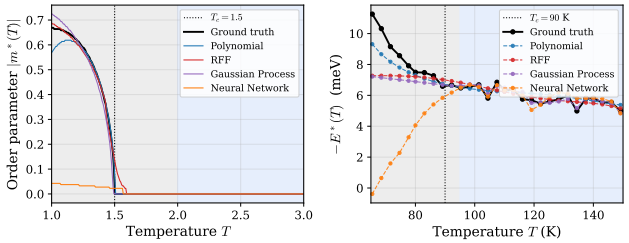


Figure 3. Recovered order parameter $|m^*(T)|$ for Heisenberg and recovered peak energy $-E^*(T)$ for ARPES. These curves are extracted from the predicted observable surfaces in Figure 2.

Vargas-Hernández et al. (2018) in two important respects. First, they show that extrapolation across phase transitions is not specific to kernel regression only, since we show that polynomial regression exhibits similar capability. Second, whereas Vargas-Hernández et al. (2018) focus on simulated data, our results show that this behavior extends to noisy experimental data.

The empirical transfer coefficient depends on the trained model and can change with the number of labeled training samples N. As N increases, the validation error on P may decrease at a different rate than the extrapolation error on Q. By contrast, for a fixed feature map, the GEVP bound is determined only by the feature covariance geometry of P and Q, which we estimate from a set number of unlabeled samples from the known domains. For polynomial regression, this sample-aware bound is far tighter than the Volume and Remez bounds, which are many orders of magnitude above the empirical transfer coefficient.

For RFF regression, the same finite-feature GEVP compu-

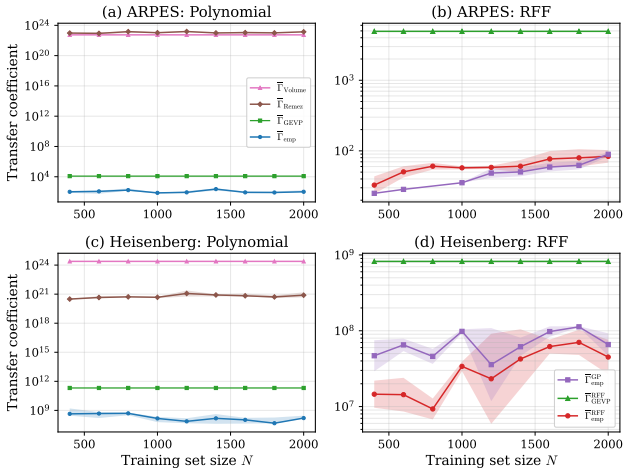


Figure 4. Transfer coefficients and bounds as the training-set size N varies. Curves show averages across 5 seeds and shaded regions show one standard deviation.

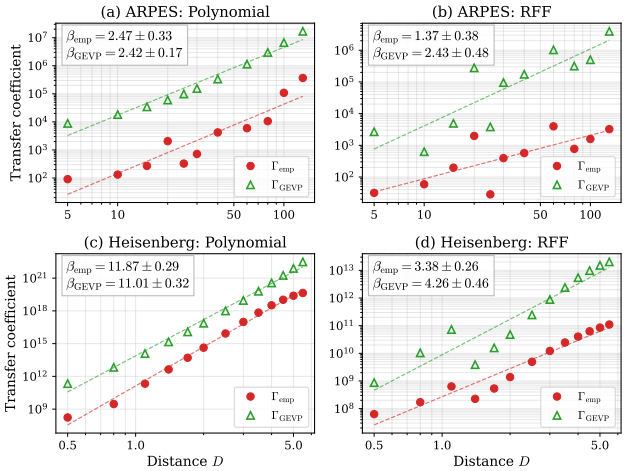


Figure 5. Empirical and GEVP transfer coefficients (log scale) versus extrapolation distance D (log scale) at fixed training size $N = 2000$. Both the empirical transfer coefficient and the GEVP bound grow approximately as power laws in the distance D, defined as the gap between the lower-temperature end of the training region and the transition point T_c.

tation remains above the empirical RFF transfer coefficient and tracks its scale. The GP empirical transfer behavior closely tracks the RFF curve, suggesting that the finite RFF representation captures the relevant GP transfer behavior in these experiments. Combined with the results in Figure 1, these findings indicate that the RFF empirical transfer coefficient is a robust proxy for the GP empirical transfer coefficient across number of features and the number of training samples. By extension, the RFF GEVP then also works as a heuristic to upper bound both the RFF and GP empirical transfer coefficient.

Importantly, our finding that the bound remains roughly constant as a function of the training set size N suggests that both polynomial and kernel regression can serve as practical tools for extrapolation in practical research settings in which researchers may not have access to large data samples.

3: Scaling Law of Extrapolation

The transfer coefficient follows a power-law scaling in the gap distance separating the training region from the transition point. Extrapolation therefore becomes harder in a structured way. This holds for both the empirical transfer coefficient and the GEVP bound. Surprisingly, the extrapolation error does not blow up rapidly (e.g. exponentially) as the distance to non-analytic behavior increases.

As the distance D between the training region and the extrapolation region grows, extrapolation becomes harder, as the model must correctly extrapolate across a gap of size D to capture the phase change. Surprisingly, as Figure 5 shows, both the GEVP bound and the empirical transfer coefficient do not grow rapidly, e.g., exponentially, but instead follow a scaling law as a function of this distance. We note, however, that the relevant distance scale is short, reflecting the small scale of the data; this dependence may therefore change at larger distances outside the support of our data.

Nonetheless, this observation is important because it suggests that extrapolation across large gaps may be possible as the bound worsens only relatively slowly. This provides further evidence that extrapolation may be feasible in more practical settings, where the distance to the transition point, if such a point exists, is unknown.

5. Conclusion

In this work, we provide bounds on extrapolation across phase transitions, taking a first step toward a principled framework for extrapolation in theoretical physics. We show that polynomial regression and kernel methods, in contrast to the neural network architectures we test, can extrapolate across a phase boundary and recover the qualitative behavior of the unobserved phase. We demonstrate this on the solvable Heisenberg model of ferromagnetism and experimental ARPES data for a high-T_c superconductor. To quantify extrapolation reliability, we study the transfer coefficient, which converts validation error on the training region into a bound on error in the extrapolation region. Among the bounds we consider, the GEVP bound is the tightest because it uses the specific feature covariance geometry of the training and extrapolation domains, rather than only coarse information such as their relative volumes or an enclosing convex region. This yields a substantial improvement over the Volume and Remez bounds, which are problem-agnostic

geometric bounds and therefore remain loose in our setting. Empirically, the GEVP bound remains informative as the number of labeled training samples varies, showing that the bound is not tied to a single training-set size. A key additional finding is that extrapolation difficulty follows a structured scaling law. These results suggest that physically motivated finite-dimensional hypothesis classes can make extrapolation across continuous phase transitions not only empirically successful, but also quantitatively bounded. More broadly, they support the possibility of reliable extrapolation methods for theoretical physicists seeking to predict physical behavior in regimes where direct data are limited, inaccessible, or not yet known.

Limitations. The main limitation of the GEVP approach is that it requires the residual $r = f^* - \hat{f}$ to lie in a specified finite-dimensional hypothesis class. For the Heisenberg target, this assumption is physically motivated by Landau theory, which directly suggests a low-order polynomial expansion. For ARPES, the assumption is weaker. Instead, we argue that the behavior across the transition can be captured by a polynomial expansion as an empirical approximation to a smooth experimental observable away from the critical point. A second limitation is the connection between RFFs and Gaussian processes. The GEVP bound is formal for the finite RFF model, but its use as a proxy for the GP is empirical, since our Gaussian RFF features do not exactly match the iteratively constructed compositional GP kernel. Finally, our experiments focus on continuous phase transitions. While we expect that expert physical constraints may allow similar extrapolation bounds to be useful in other physics settings, extending to other phenomena remains far from guaranteed.

Impact Statement

The main societal implication is more reliable extrapolation tools could help scientists reason about physical regimes where data are limited or expensive to obtain.

References

Aronszajn, N. Theory of reproducing kernels. *Transactions of the American Mathematical Society*, 68(3):337–404, 1950. doi: 10.1090/S0002-9947-1950-0051437-7.

Bach, F. On the equivalence between kernel quadrature rules and random feature expansions. *Journal of Machine Learning Research*, 18(21):1–38, 2017.

Brudnyi, Y. A. and Ganzburg, M. I. On an extremal problem for polynomials in n variables. *Math. USSR-Izvestiya*, 7: 345–356, 1973.

Carrasquilla, J. and Melko, R. G. Machine learning phases of matter. *Nature Physics*, 13(5):431–434, 2017.

Duvenaud, D., Lloyd, J. R., Grosse, R., Tenenbaum, J. B., and Ghahramani, Z. Structure discovery in nonparametric regression through compositional kernel search. In *International Conference on Machine Learning*, 2013.

Gerlach, B. and Löwen, H. Analytical properties of polaron systems or: Do polaronic phase transitions exist or not? *Reviews of Modern Physics*, 63(1):63–90, 1991. doi: 10.1103/RevModPhys.63.63.

Grosse, R., Salakhutdinov, R., Freeman, W. T., and Tenenbaum, J. B. Exploiting compositionality to explore a large space of model structures. In *Uncertainty in Artificial Intelligence (UAI)*, 2012.

He, Y., Chen, S.-D., Li, Z.-X., Zhao, D., Song, D., Yoshida, Y., Eisaki, H., Wu, T., Chen, X., Lu, D., Meingast, C., Devereaux, T. P., Birgeneau, R. J., Hashimoto, M., Lee, D.-H., and Shen, Z.-X. Superconducting fluctuations in overdoped $Bi_2Sr_2CaCu_2O_{8+\delta}$. *Physical Review X*, 11(3): 031068, 2021.

Herrera, F., Madison, K. W., Krems, R. V., and Berciu, M. Investigating polaron transitions with polar molecules. *Physical Review Letters*, 110(22):223002, 2013. doi: 10.1103/PhysRevLett.110.223002.

Kalavasis, A., Zadik, I., and Zampetakis, M. Transfer learning beyond bounded density ratios, 2024. Preprint.

Landau, L. D. and Lifshitz, E. M. *Statistical Physics, Part 1*, volume 5 of *Course of Theoretical Physics*. Pergamon Press, 3 edition, 1980.

Lau, B., Berciu, M., and Sawatzky, G. A. Single-polaron properties of the one-dimensional breathing-mode hamiltonian. *Physical Review B*, 76(17):174305, 2007. doi: 10.1103/PhysRevB.76.174305.

Lloyd, J. R., Duvenaud, D., Grosse, R., Tenenbaum, J. B., and Ghahramani, Z. Automatic construction and natural-language description of nonparametric regression models. In *AAAI Conference on Artificial Intelligence*, 2014.

Marchand, D. J. J., De Filippis, G., Cataudella, V., Berciu, M., Nagaosa, N., Prokof'ev, N. V., Mishchenko, A. S., and Stamp, P. C. E. Sharp transition for single polarons in the one-dimensional su-schrieffer-heeger model. *Physical Review Letters*, 105(26):266605, 2010. doi: 10.1103/PhysRevLett.105.266605.

Rahimi, A. and Recht, B. Random features for large-scale kernel machines. In *Advances in Neural Information Processing Systems*, 2007.

Rasmussen, C. E. and Williams, C. K. I. *Gaussian Processes for Machine Learning*. MIT Press, 2006.

Rudi, A. and Rosasco, L. Generalization properties of learning with random features. In *Advances in Neural Information Processing Systems*, 2017.

Schölkopf, B. and Smola, A. J. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press, Cambridge, MA, 2002.

Schwarz, G. Estimating the dimension of a model. *The Annals of Statistics*, 6(2):461–464, 1978. doi: 10.1214/aos/1176344136.

Sitzmann, V., Martel, J. N. P., Bergman, A. W., Lindell, D. B., and Wetzstein, G. Implicit neural representations with periodic activation functions. In *Advances in Neural Information Processing Systems*, 2020.

Sutherland, D. J. and Schneider, J. On the error of random fourier features. In *Conference on Uncertainty in Artificial Intelligence*, 2015.

Tancik, M., Srinivasan, P. P., Mildenhall, B., Fridovich-Keil, S., Raghavan, N., Singhal, U., Ramamoorthi, R., Barron, J. T., and Ng, R. Fourier features let networks learn high frequency functions in low dimensional domains. In *Advances in Neural Information Processing Systems*, 2020.

van Nieuwenburg, E. P. L., Liu, Y.-H., and Huber, S. D. Learning phase transitions by confusion. *Nature Physics*, 13:435–439, 2017.

Vargas-Hernández, R. A., Sous, J., Berciu, M., and Krems, R. V. Extrapolating quantum observables with machine learning: Inferring multiple phase transitions from properties of a single phase. *Physical Review Letters*, 121(25): 255702, 2018.

Wang, L. Discovering phase transitions with unsupervised learning. *Physical Review B*, 94:195105, 2016.

Wetzel, S. J. Unsupervised learning of phase transitions: From principal component analysis to variational autoencoders. *Physical Review E*, 96:022140, 2017.

Wilson, A. G. and Adams, R. P. Gaussian process kernels for pattern discovery and extrapolation. In *International Conference on Machine Learning*, 2013.

Xu, K., Zhang, M., Li, J., Du, S. S., Kawaguchi, K.-i., and Jegelka, S. How neural networks extrapolate: From feedforward to graph neural networks. In *International Conference on Learning Representations (ICLR)*, 2021.

Appendix

A. Derivations

Volume bound. Kalavasis et al. (2024) prove a disjoint-support transfer bound for degree-d polynomials. Let $P, Q \subset R_{\text{full}} \subset \mathbb{R}^n$ be disjoint rectangles equipped with uniform measure, where n is the input dimension, and let $r \in \mathbb{P}_d$. Define the inverse relative volumes

$$\rho_P = |R_{\text{full}}|/|P|, \qquad \rho_Q = |R_{\text{full}}|/|Q|.$$

Then

$$\mathrm{MSE}_Q(r) \le (2d)^{2d}\, \rho_Q\, \rho_P^{2d}\, \mathrm{MSE}_P(r).$$

The proof uses a Bernstein-type estimate for polynomials on the unit cube, followed by an affine change of variables from a generic rectangle to $[-1, 1]^n$. The important point for our experiments is that this bound is geometry-based.

Remez bound. The multivariate Remez inequality (Brudnyi & Ganzburg, 1973) controls the maximum value of a polynomial on a convex set from its maximum value on a measurable subset. Let $K \subset \mathbb{R}^n$ be convex, let $P \subset K$ have relative volume $\lambda = |P|/|K|$, and let $p \in \mathbb{P}_d$. Then

$$\sup_{x \in K} |p(x)| \le \left(\frac{4n}{\lambda}\right)^d \sup_{x \in P} |p(x)|.$$

Apply this to the residual $r \in \mathbb{P}_d$ with $K = R_{\text{full}}$ and $\varepsilon = \sup_{x \in P} |r(x)|$. Squaring gives the pointwise bound

$$r(x)^2 \le \left(\frac{4n}{\lambda}\right)^{2d} \varepsilon^2 \qquad \forall x \in K.$$

Since $Q \subset K$, averaging this pointwise bound over Q immediately gives

$$\mathrm{MSE}_Q(r) = \frac{1}{|Q|} \int_Q r(x)^2\, dx \le \left(\frac{4n}{\lambda}\right)^{2d} \varepsilon^2.$$

In our experiments $n = 2$. For example, when $\lambda = 0.5$ and $d = 8$, the prefactor is

$$\left(\frac{4n}{\lambda}\right)^{2d} = \left(\frac{8}{0.5}\right)^{16} = 16^{16} = 2^{64} \approx 1.84 \times 10^{19}.$$

Thus, even in two input dimensions, the Remez bound is useful mainly as a worst-case sanity check rather than as a tight bound of the empirical transfer coefficient.

GEVP derivation. The GEVP bound is the finite-feature analogue of the same transfer question, but it uses the actual feature covariance geometry of the training and extrapolation regions. Let $\phi : \mathcal{X} \to \mathbb{R}^k$ be a finite feature map and let $g_c(x) = c^\top \phi(x)$ be any function in its linear span. Define

$$B = \mathbb{E}_{x \sim P}[\phi(x)\phi(x)^\top], \qquad A = \mathbb{E}_{x \sim Q}[\phi(x)\phi(x)^\top].$$

Then the squared error of g_c on the two regions is

$$\mathbb{E}_{x \sim P}[g_c(x)^2] = c^\top B c, \qquad \mathbb{E}_{x \sim Q}[g_c(x)^2] = c^\top A c.$$

Therefore the transfer ratio of this feature-space direction is the generalized Rayleigh quotient

$$\frac{\mathbb{E}_{x \sim Q}[g_c(x)^2]}{\mathbb{E}_{x \sim P}[g_c(x)^2]} = \frac{c^\top A c}{c^\top B c}, \qquad c \ne 0.$$

When B is positive definite on the feature span, the Rayleigh–Ritz principle gives

$$\sup_{c \ne 0} \frac{c^\top A c}{c^\top B c} = \lambda_{\max}(A, B),$$

where $\lambda_{\max}(A, B)$ is the largest generalized eigenvalue of the pair (A, B), attained by a vector v satisfying $Av = \lambda_{\max} Bv$.

This is the key intuition used in the main text: the GEVP searches for the feature-space direction that is small on P but large on Q. It is therefore the worst-case amplification factor from the training region to the extrapolation region for the chosen finite feature class. If the realized residual $r = f^* - \hat{f}$ lies in the same finite feature span, then substituting $g_c = r$ into the worst-case ratio gives

$$\mathrm{MSE}_Q(\hat{f}) \leq \lambda_{\max}(A, B)\, \mathrm{MSE}_P(\hat{f}).$$

Using $\phi = \phi_d$ gives the polynomial GEVP bound, and using $\phi = z$ gives the RFF GEVP bound.

In practice, A and B are estimated using large unlabeled Monte Carlo samples from the known regions Q and P. These samples are not the labeled training data used to fit $\hat{f}$. For a fixed feature map, the population GEVP coefficient depends on the feature representation and the geometry of P, Q, not on the number of labeled training samples.

Numerically, we compute a regularized generalized eigenvalue problem via `scipy.linalg.eigh`$(A, B + \epsilon I)$, with ϵ chosen for numerical stability. This regularization prevents nearly null directions of B from producing unstable eigenvalues. Because adding ϵI to the denominator matrix increases the denominator of the generalized Rayleigh quotient, the computed value is a regularized approximation to the unregularized transfer coefficient. We use this stabilized value in the experiments and interpret it as the practical finite-feature bound.

B. Random-Feature Approximation Background

This appendix explains why random Fourier features are a natural finite-dimensional comparison point for Gaussian processes, and also clarifies the limits of this connection. The main point is that the RFF GEVP is an exact bound for the finite RFF feature class. It is not a formal finite-m upper bound on the full Gaussian-process kernel.

Bochner's theorem and random Fourier features. Let $k(x, y) = \kappa(x - y)$ be a continuous translation-invariant positive-definite kernel on $\mathbb{R}^n$. By Bochner's theorem, κ is the Fourier transform of a non-negative finite measure τ:

$$\kappa(x - y) = \int_{\mathbb{R}^n} e^{i\omega^\top (x-y)}\, d\tau(\omega).$$

Random Fourier features approximate this integral by Monte Carlo sampling. Drawing $\omega_j \sim \tau$ and $b_j \sim \mathrm{Uniform}(0, 2\pi)$ gives features

$$z_j(x) = \sqrt{2/m} \cos(\omega_j^\top x + b_j), \qquad j = 1, \ldots, m,$$

for which $\mathbb{E}[z(x)^\top z(y)] = k(x, y)$ (Rahimi & Recht, 2007). Thus RFFs replace a kernel evaluation by an inner product in a finite random feature space.

What Bach's approximation results imply. Bach (2017) studies when finite random-feature expansions approximate the function class associated with a positive-definite kernel. In that framework, the random features are sampled from a feature distribution q, and the approximation rate depends on an effective dimension quantity, often written as $d_{\max}(q, \lambda)$, at regularization level λ. Under the assumptions of the theorem, a sufficiently large number of random features gives a high-probability approximation of the kernel RKHS in $L^2(\rho)$, where ρ is the input distribution. Related work gives learning guarantees for random-feature regression under comparable regularity and sampling assumptions (Rudi & Rosasco, 2017; Sutherland & Schneider, 2015).

These results motivate using RFFs as finite-dimensional approximations to kernel methods. They do not, by themselves, imply that the unregularized transfer coefficient of the full RKHS is finite. The transfer coefficient involves the ratio

$$\sup_f \frac{\mathbb{E}_{x\sim Q}[f(x)^2]}{\mathbb{E}_{x\sim P}[f(x)^2]}.$$

For a rich or universal RKHS on disjoint regions, this unregularized ratio can be ill-conditioned or unbounded because one can construct functions that are small on P and large on Q. Equivalently, the covariance operator associated with P may be compact and may not have a bounded inverse, so the corresponding generalized eigenvalue problem is not a well-conditioned finite-dimensional limit.

How we use this in our experiments. Our use of RFFs is therefore deliberately finite-dimensional. For a fixed number of random features m, the RFF feature map defines a finite function class $\mathcal{G}_m$, and the GEVP bound in Proposition 3.4 is exact for that finite class, subject to the residual-realizability assumption stated in the proposition. This finite-dimensional truncation is what makes the generalized eigenvalue computation well-defined.

The connection to Gaussian processes is empirical in our experiments. In the matched setting, where the random features are sampled from the same spectral measure as the kernel, Bach's results support the intuition that increasing m makes the random-feature model approximate the corresponding kernel method. In our experiments, however, the RFF features are sampled from a Gaussian spectral measure, while the GP uses an iteratively constructed compositional kernel. The formal assumptions needed to identify the RFF feature class with the GP RKHS are therefore not satisfied. We consequently interpret Figure 1 as empirical evidence that the finite RFF transfer behavior tracks the GP transfer behavior in our tested regimes, not as a theorem that the RFF GEVP upper bounds the full GP transfer coefficient.

Finally, the numerical GEVP computation uses regularization of the denominator matrix for stability, as described in Appendix A. This regularization further reinforces the finite-dimensional interpretation: the reported RFF GEVP is an empirical bound for the stabilized finite-feature problem, not an unregularized infinite-dimensional GP bound.

C. Datasets

Heisenberg. We construct the mean-field Landau–Ginzburg free energy dataset of (1) with $T_c = 1.5$ and an additive offset $c_0 = 0.01$. The training and target rectangles are $P = [2.0, 3.0] \times [-1.25, 1.25]$ and $Q = [1.0, 2.0] \times [-1.25, 1.25]$, so $\rho_P = \rho_Q = 2$. Each training set is drawn uniformly on P at the chosen seed; a fixed 2000-point uniform set on Q at seed 99 is held aside as the evaluation grid and is shared identically across all methods, so any difference in extrapolation MSE comes from the model rather than the test sample.

ARPES Bi-2212. Antinodal energy-distribution curves of Pb-doped Bi-2212 from He et al. (2021), with $T_c \approx 90\,\mathrm{K}$. We use the central of three momentum-integration windows provided in the given data. The raw (T, E) grid has 100 temperatures non-uniformly spaced from 13.5 K to 290 K and 511 binding energies $E_j = -0.371173 + 0.001\,j$ eV; ground truth is a `RegularGridInterpolator` with linear interpolation on this grid, and NaN values outside the analysis window are replaced with column medians. Training and target domains are $P = (95, 150)\,\mathrm{K} \times (-0.1, 0.0)\,\mathrm{eV}$ and $Q = (65, 95)\,\mathrm{K} \times (-0.1, 0.0)\,\mathrm{eV}$. Samples are stratified in T across $\sqrt{N}$ uniform strata and drawn uniformly in E, which keeps the temperature axis well-covered at small N.

Training-size sweep (Figure 4). For each method, dataset, and seed in $\{42, 43, 44, 45, 46\}$ we train on 9 linearly-spaced training sizes $N \in \{400, 600, \ldots, 2000\}$. Hyperparameters selection is done on the 5 seeds for $N = 2000$ using the 80/20 holdout, and the most commonly chosen HP configuration is used to train across all (N, seed) combination. This way we completely isolate the effect of the number of training points.

Distance sweep (Figure 5). We slide the lower edge of the training rectangle $T_{\mathrm{interp,lo}}$ upward away from T_c by distance $D = T_{\mathrm{interp,lo}} - T_c$, holding the rectangle width fixed (1.0 on Heisenberg, 55 K on ARPES) so that the training density at fixed N is constant across D. The target rectangle Q stays pinned to its closest-to-T_c default. The single varying quantity is therefore the gap from T_c to the training region, isolating geometry from training density. On Heisenberg we sweep $D \in \{0.5, 0.8, 1.1, 1.4, 1.7, 2.0, 2.5, 3.0, 3.5, 4.0, 4.5, 5.0, 5.5\}$ at fixed $N = 2000$ and 5 seeds per cell. On ARPES we sweep $D \in \{5, 10, 15, 20, 25, 30, 40, 60, 80, 100, 130\}$ K under the same protocol.

D. Neural Network Methods

We benchmark two neural network variants designed against the spectral bias that hampers vanilla MLPs on high-frequency targets. We first test a Fourier-feature MLP (Tancik et al., 2020), which prepends a fixed sinusoidal embedding $\gamma(x) = (\cos(2\pi Bx), \sin(2\pi Bx))$ to a standard MLP so the network can represent high-frequency content from the input. We also test SIREN (Sitzmann et al., 2020), which replaces every activation with $\sin(\omega_0 \cdot)$ so the network is analytic with sinusoidal derivatives at all orders. Both are widely-used implicit-neural-representation baselines and were the most plausible neural candidates we knew of to extrapolate non-monotonically below T_c.

Empirically, they do not extrapolate well. Table 2 reports their extrapolation metrics alongside the vanilla MLP, and Figure 6

shows the corresponding landscape comparison: on Heisenberg both neural variants produce a worse below-T_c surface than the vanilla MLP, with FF-MLP introducing high-frequency artifacts and SIREN's sinusoidal hypothesis driving the prediction to a strongly-oscillating shape that bears no resemblance to the double-well structure. On ARPES the FF-MLP is closer to the GP extrapolation quality than the vanilla MLP but still misses the gap migration, while SIREN stays close to its training-distribution mean. The qualitative point is that the spectral-bias remedies engineered for high-frequency interpolation do not translate to good extrapolation across a phase transition.

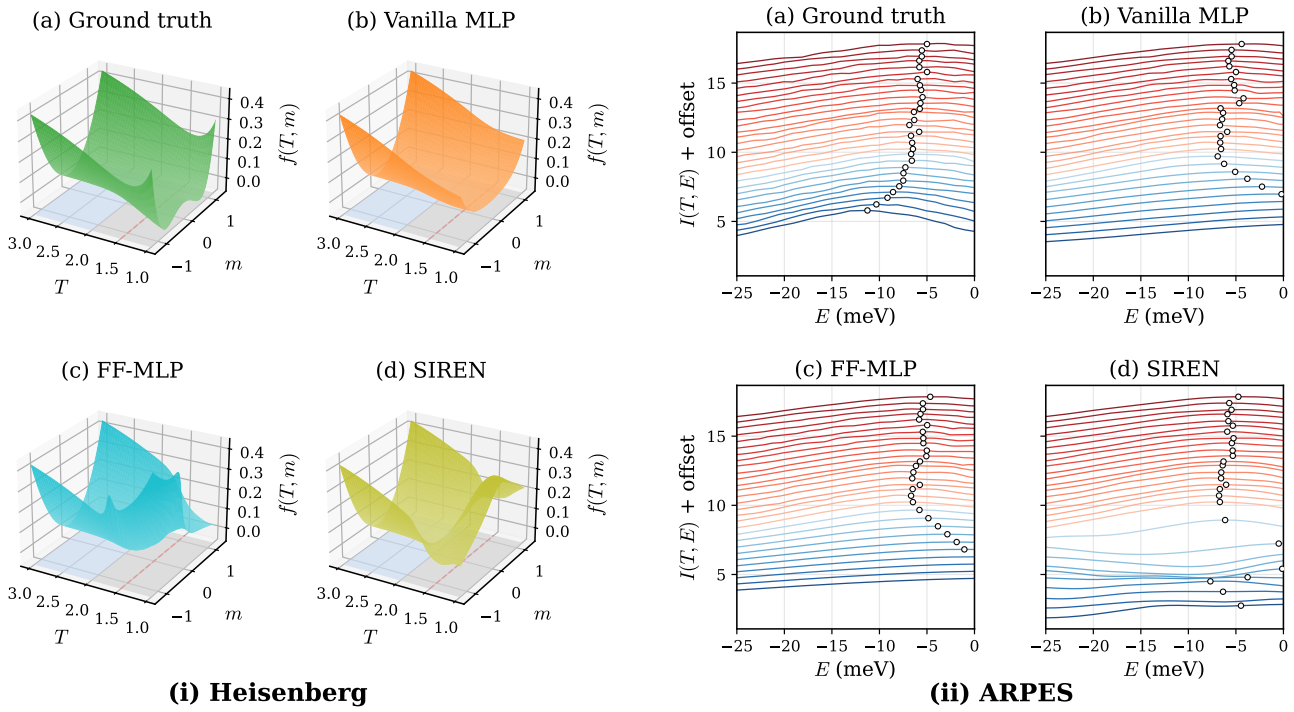


Figure 6. Neural network landscape comparison on the extrapolation domain Q for Heisenberg (i) and ARPES (ii).

Table 2. Neural network results on the same protocol as Table 1. Mean across 5 seeds at $N = 2000$.

	Heisenberg			ARPES	
Method	Extrap NMSE	Extrap Rel. %	$\lvert\Delta T_c\rvert$	Extrap NMSE	Extrap Rel. %
Vanilla MLP	2.0×10^{-1}	28.8	—	1.1×10^{-1}	23.1
FF-MLP	4.2×10^{0}	117.6	0.334	9.4×10^{-2}	19.6
SIREN	5.3×10^{0}	168.0	0.300	1.1×10^{0}	79.0

E. Training Details

Training details and hyperparameter grids by method are listed below, the resulting selections are summarized in Table 3.

Polynomial. Grid: degree $d \in \{2, 4, 6, 8\}$, ridge regularization fixed at $\alpha = 10^{-6}$ on both datasets. On Heisenberg we replace T by $1/T$ before the expansion. Only the degree is selected, by type-II maximum log marginal likelihood.

RFF. Grid: $m \in \{1000, 2000, 4000, 8000\}$, anisotropic length-scale factors $\{0.25, 0.5, 1, 2, 4\}$ per axis (median pairwise distance times factor), ridge $\alpha \in \{10^{-8}, 10^{-6}, 10^{-4}, 10^{-2}\}$. Log marginal likelihood is averaged over 3 RFF draws to stabilize selection.

GP. `scikit-learn`'s `GaussianProcessRegressor` with `normalize_y=True`. Base kernels $\{\mathrm{LIN}, \mathrm{RBF\text{-}ARD}, \mathrm{MAT5/2\text{-}ARD}, \mathrm{RQ}\} \otimes \mathrm{ConstantKernel}$, anisotropic length scales throughout. Each candidate kernel is fit by maximum log marginal likelihood (`n_restarts_optimizer=3` L-BFGS restarts for the depth

winner, 1 for the candidate scan). The depth-wise greedy search ranks candidates by BIC and accepts the winner; depth ≤ 3. Fixed noise $\alpha = 10^{-8}$ for both datasets.

MLP. Residual fully-connected with LayerNorm. Grid: depth $\in \{2, 4, 6, 8\}$, width $\in \{8, 128, 256, 512\}$, activation $\in \{\text{GELU}, \tanh\}$, learning rate $\in \{10^{-2}, 3 \times 10^{-3}\}$, weight decay $\in \{10^{-5}, 10^{-4}\}$. AdamW with early stopping on the 20% holdout of P; max 5×10^5 epochs and patience 500. ReLU MLPs were excluded after a pilot sweep showed high validation loss across the same grid.

FF-MLP (Tancik et al., 2020). Same depth/width/learning-rate/weight-decay grid as the vanilla MLP, plus the Fourier-feature embedding $\gamma(x) = (\cos(2\pi Bx), \sin(2\pi Bx))$ with $B \in \mathbb{R}^{m_{\text{ff}} \times n}$ drawn i.i.d. from $\mathcal{N}(0, \sigma^2)$; we tune $m_{\text{ff}} \in \{64, 128, 256\}$ and $\sigma \in \{0.05, 0.1, 0.5, 1.0\}$.

SIREN (Sitzmann et al., 2020). Sinusoidal activations $\sin(\omega_0 \cdot)$ at every layer; we tune $\omega_{0,\text{first}} \in \{2, 5, 10\}$ for the first-layer frequency and $\omega_{0,\text{hidden}} \in \{1, 5, 10\}$ for the hidden layers, with the same depth/width grid as the MLP and a learning rate sweep $\{10^{-4}, 10^{-3}\}$ at zero weight decay.

Compute. Polynomial, RFF, and GP training and the GEVP/Volume/Remez bound computations ran on CPU partitions (4–16 cores per job, 16–64 GB RAM per job, jobs typically minutes per cell on the largest training-size sweep cell). Neural network training (vanilla MLP, FF-MLP, SIREN) ran on the H200 GPU partitions, and took typically ten minutes per seed. The total compute budget for the experiments reported in this paper, including hyperparameter selection, training-size and distance sweeps, and the Bach feature-count sweep, was on the order of a few 100 CPU-hours and GPU-hours.

Table 3. Hyperparameters selected on the 80/20 holdout at $n = 2000$ for each method on each dataset.

Method	Heisenberg	ARPES
Polynomial	$d = 8$, basis $(1/T, m)$, $\alpha = 10^{-6}$	$d = 8$, basis (T, E), $\alpha = 10^{-6}$
RFF	$m = 8000$, $\ell = (3.58, 0.45)$, $\alpha = 10^{-8}$	$m = 2000$, $\ell = (3.53, 0.44)$, $\alpha = 10^{-4}$
GP	$((\text{RBF} \times \text{LIN}) \times \text{LIN}) \times \text{LIN}$, anisotropic ℓ, $\alpha = 10^{-8}$	$\text{MAT}5/2 + \text{RBF}$, anisotropic ℓ, $\alpha = 10^{-8}$
MLP	$[256, 256]$, GELU, $\eta = 3 \times 10^{-3}$, wd $= 10^{-5}$	$[512{\times}4]$, GELU, $\eta = 3 \times 10^{-3}$, wd $= 10^{-4}$
FF-MLP	$m_{\text{ff}} = 128$, $\sigma = 0.1$, $[256{\times}4]$, GELU, $\eta = 3 \times 10^{-3}$, wd $= 10^{-4}$	$m_{\text{ff}} = 256$, $\sigma = 0.05$, $[256{\times}6]$, GELU, $\eta = 10^{-3}$, wd $= 10^{-4}$
SIREN	$[256{\times}4]$, $\omega_{0,\text{first}} = 2$, $\omega_{0,\text{hidden}} = 1$, $\eta = 10^{-4}$	$[256{\times}4]$, $\omega_{0,\text{first}} = 10$, $\omega_{0,\text{hidden}} = 5$, $\eta = 10^{-4}$

F. Supplementary Figures

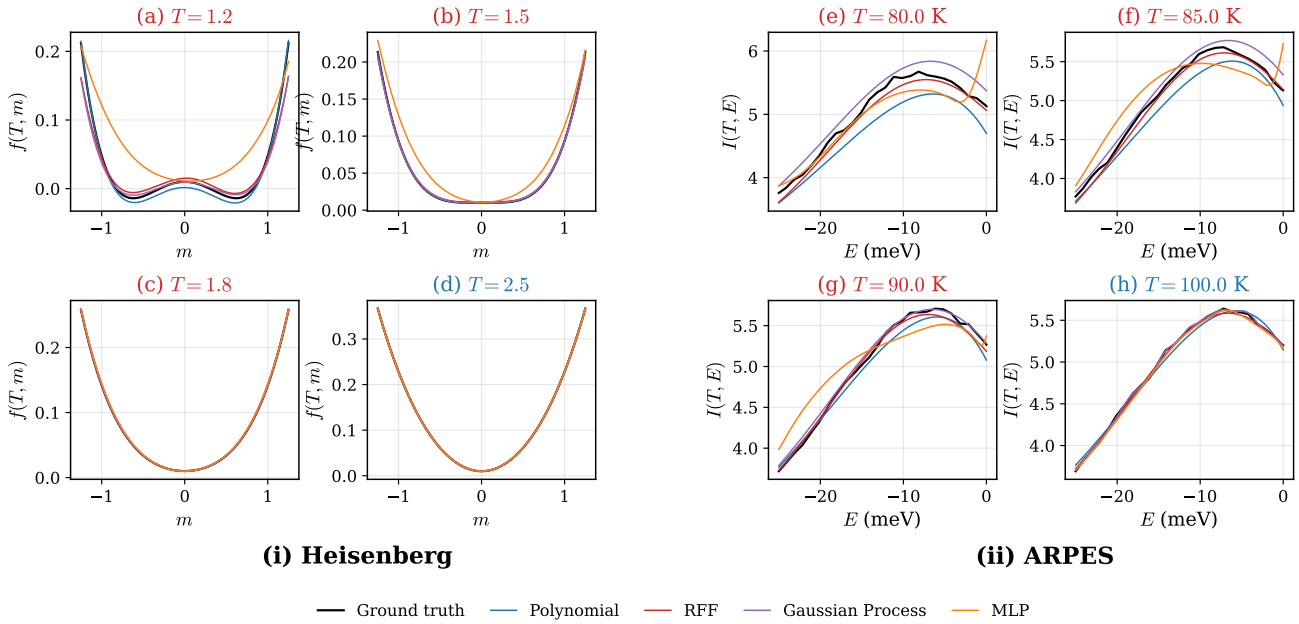


Figure 7. Slices at four fixed temperatures spanning T_c for Heisenberg (i) and ARPES (ii). Generated from taking temperature slices of Figure 2.