Dimension independent data sets approximation and applications to classification

Patrick Guidotti

Guidotti *Advanced Modeling and Simulation in Engineering Sciences(2024)11:1*
https://doi.org/10.1186/s40323-023-00256-w

Advanced Modeling and Simulation in Engineering Sciences

RESEARCH ARTICLE Open Access

Dimension independent data sets approximation and applications to classification

Patrick Guidotti

*Correspondence: gpatrick@math.uci.edu
[1]Department of Mathematics, University of California, 340 Rowland Hall, Irvine, CA 92697-3875, USA

Abstract

We revisit the classical kernel method of approximation/interpolation theory in a very specific context from the particular point of view of partial differential equations. The goal is to highlight the role of regularization by casting it in terms of actual smoothness of the interpolant obtained by the procedure. The latter will be merely continuous on the data set but smooth otherwise. While the method obtained fits into the category of RKHS methods and hence shares their main features, it explicitly uses smoothness, via a dimension dependent (pseudo-)differential operator, to obtain a flexible and robust interpolant, which can adapt to the shape of the data while quickly transitioning away from it and maintaining continuous dependence on them. The latter means that a perturbation or pollution of the data set, small in size, leads to comparable results in classification applications. The method is applied to both low dimensional examples and a standard high dimensioal benchmark problem (MNIST digit classification).

Keywords: Kernel based interpolation, Supervised classification, Data analysis

Introduction

The problem of finding a function that explains a given set of data is a fundamental problem in mathematics and statistics. If the data are assumed to be the discrete manifestation of a function defined on a continuous (as opposed to discrete) domain of definition, the problem can be viewed as an approximation problem where the data can be leveraged to help identify a sensible approximation to the function. Often one resorts the prior knowledge about the target function to reduce the set of candidates from which to choose a good approximation, if not the best approximation. Within this framework, an extensive mathematical knowledge has been obtained over the past several decades along with a variety of powerful tools (see [1], in particular, for the philosophical approach taken here). In the current world, where data about almost anything you can imagine or wish for is available, one of the most interesting and often challening problems consists in extracting information, knowledge, and structure from high dimensional data. A variety of commonly used approaches belong to a category referred to as Machine Learning. Neural networks in all forms and shapes are particularly widespread due to their success in dealing with a

series of challenging problems. Another class is that of Support-Vector Machines (SVM) [2], which were very popular before being somewhat superseded by improved neural networks. While they often use linear classification via hyperplanes, the so-called kernel trick [3,4] makes it possible for them to capture nonlinear decision boundaries albeit by working in a higher dimensional (feature) space to which the data is mapped but that admits an efficient computation of scalar products (via a suitable kernel). A connection between SVMs and neural networks was discovered in [5] and has since been investigated by many more authors. The method proposed here is philosophically in the category of SVMs but distinguishes itself by directly working with the data at hand without embedding into a higher dimensional feature space. This is made possible by directly looking for a good approximation in a large space of functions that allows for nonlinear behavior as opposed to a priori restricting the space of possible approximants by choosing a feature vector that replaces the data and on which the eventual approximant depends linearly upon. It will turn out that the end result of the approach taken here bears similarity with the

We first review a classical method of inexact interpolation that yields a continuous function approximating a given data set. We do so by taking a PDE perspective that reveals important features that are exploited in order to obtain the announced stable method of classification even when the distribution of available data is not uniform in space and, possibly, noisy. In high dimension, even large data sets are often sparse due to the so-called curse of dimensionality. Moreover, the data is often supported on lower dimensional manifolds, where even dense data make up a thin slice of the ambient space. In the latter case, it is demonstrated in this paper that, while the data may not be sufficient for the reliable identification of a well-defined global approximant, it can still be used fruitfully (in a global or local fashion) for data analysis purposes. This is mainly due to the fact that the proposed method is capable of connecting the dots (data points) into a manifold by capturing geometric features of the data.

It is widely understood and accepted that a function interpolating discrete data should be at least continuous so as to provide a certain stability of prediction and resilience in the face of noise. In learning problems this is sometimes expressed as local constancy, even if that does not require continuity. Unless the data is known to stem from a very smooth underlying function, but, typically, even in that case, it should also not be exceedingly smooth, as this would lead to some blurring and reduce its ability to capture sharp transitions. It would, moreover, be advantageous for the interpolating function to depend at least continuously on the data set it is constructed from as, in that case, perturbations (due to measurement errors or to other sources) would not have a large impact on the outcome of classification based on the inperpolant. This kind of stability is akin to that discussed in [6] in the context of learning algorithms.

The starting point is a set of data consisting of point/value pairs

$$\mathbb{D} = \left\{(x^i, y^i) \,\middle|\, i = 1, \ldots, m\right\}$$

where $m \in \mathbb{N}$, $x^i \in \mathbb{R}^d$ and $y^i \in \mathbb{R}^n$, $i = 1, \ldots, m$, for $d, n \in \mathbb{N}$. In order to enforce minimal regularity on the interpolant function

$$u : \mathbb{R}^d \to \mathbb{R}^n$$

we take it from the space

$$\mathrm{H}^{\frac{d+1}{2}}(\mathbb{R}^d, \mathbb{R}^n) = \left\{u \in \mathcal{S}' \,\middle|\, [\xi \mapsto (1 + |\xi|^2)^{\frac{d+1}{4}} \hat{u}(\xi)] \in \mathrm{L}^2(\mathbb{R}^d, \mathbb{R}^n)\right\},$$

where $\mathcal{S}'$ denotes the Schwartz space of tempered distributions, i.e. the topological dual of the space of smooth, rapidly decreasing functions. Thanks to the general Sobolev inequality it holds that

$$\mathrm{H}^{\frac{d+1}{2}}(\mathbb{R}^d,\mathbb{R}^n)\hookrightarrow \mathrm{BUC}^{\frac{1}{2}}(\mathbb{R}^d,\mathbb{R}^n), \tag{0.1}$$

where the containing space is that of bounded and uniformly Hölder continuous functions of exponent $\frac{1}{2}$. In order to obtain stability we may sacrifice some interpolation accuracy by not necessarily requiring the exact validity of

$$u(x^i)=y^i \text{ for } i=1,\dots,m. \tag{0.2}$$

Then, for approximate interpolation, u is determined by minimization of the energy functional given by

$$E_\alpha(u)=\frac{\alpha}{2c_d}\int_{\mathbb{R}^d}\left|(1-\Delta)^{\frac{d+1}{4}}u(x)\right|^2dx+\frac{1}{2}\sum_{i=1}^m\left|u(x^i)-y^i\right|^2 \tag{0.3}$$

$$=\frac{\alpha}{2c_d}\|u\|^2_{\mathrm{H}^{\frac{d+1}{2}}}+\frac{1}{2}\sum_{i=1}^m\left|u(x^i)-y^i\right|^2,\ u\in\mathrm{H}^{\frac{d+1}{2}}(\mathbb{R}^d,\mathbb{R}^n) \tag{0.4}$$

for $\alpha>0$ and where the normalizing constant c_d will be explicitly given in the next section (right below equation (2.6)). For exact interpolation u is determined by minimization of $E_0=\|\cdot\|^2_{\mathrm{H}^{\frac{d+1}{2}}}$ with constraints (0.2) that shall be summarized as $u(\mathbb{X})=\mathbb{Y}$. Formally, it is set

$$u_{\mathbb{D},\alpha}=\begin{cases}\operatorname{argmin}_{u\in\mathrm{H}^{\frac{d+1}{2}}(\mathbb{R}^d,\mathbb{R}^n)}E_\alpha(u), & \alpha>0,\\ \operatorname{argmin}_{u\in\mathrm{H}^{\frac{d+1}{2}}(\mathbb{R}^d,\mathbb{R}^n),\,u(\mathbb{X})=\mathbb{Y}}E_0(u), & \alpha=0.\end{cases} \tag{0.5}$$

While in many practical problems the dimension d can be very large and the constant c_d astronomical, this approach remains viable since the minimizers can be identified by solving a well-posed $m\times m$ linear system of equations for any $\alpha\geq 0$. The rest of the paper is organized as follows. In the next section we provide a detailed description of the method and obtain some of its basic mathematical properties. In the following section, we discuss a variety of numerical experiments that showcase the viability and efficacy of the method.

There are interesting connections between this method and kernel based interpolation

The method

In order to derive a concrete method it needs to be shown that minimizers of E_d can be computed efficiently. We first observe that the functional has a unique minimizer no matter what the given data set is.

Theorem 2.1 *The functional E_α has a unique minimizer $u_{\mathbb{D},\alpha}$ for any given data set $\mathbb{D}$.*

Proof Take $\alpha>0$ first. Thanks to the embedding (0.1) the functional $E_\alpha:\mathrm{H}^{\frac{d+1}{2}}(\mathbb{R}^d,\mathbb{R}^n)\to[0,\infty)$ is continuous. It is clearly also strictly convex as a quadratic functional since the first term is the square of a norm. Strongly lower semi-continuous

convex functionals on a Hilbert space are known to be weakly lower-semicontinuous and so is therefore E_α. It is also coercive since closed bounded sets in $\mathrm{H}^{\frac{d+1}{2}}(\mathbb{R}^d,\mathbb{R}^n)$ are relatively weakly compact by the Banach-Alaoglu Theorem. A weakly lower-semicontinuous and coercive functional possesses a minimum, which, by strict convexity, is necessarily unique. The case $\alpha=0$ uses essentially the same argument where the full function space is replaced by the convex closed subset of functions satisfying $u(\mathbb{X})=\mathbb{Y}$. □

Now that a unique continuous interpolant $u_{\mathbb{D}}$ has been obtained for each given data set $\mathbb{D}$, it needs to be determined in a usable form. The next step consists in deriving the Euler-Lagrange equation for $u_{\mathbb{D}}$.

Theorem 2.2 *The minimizer $u_{\mathbb{D},\alpha}$ for $\alpha>0$ satisfies the equation (system)*

$$\frac{\alpha}{c_d}(1-\Delta)^{\frac{d+1}{2}}u=\sum_{i=1}^m\big[y^i-u(x^i)\big]\delta_{x^i} \tag{2.1}$$

in the weak sense, i.e. the equation holds in the space $\mathrm{H}^{-\frac{d+1}{2}}(\mathbb{R}^d,\mathbb{R}^n)$ or, equivalently it holds that

$$\frac{\alpha}{c_d}\int_{\mathbb{R}^d}\big[(1-\Delta)^{\frac{d+1}{4}}u\big](x)\cdot\big[(1-\Delta)^{\frac{d+1}{4}}v\big](x)\,dx=\sum_{i=1}^m\big[y^i-u(x^i)\big]\cdot v(x^i), \tag{2.2}$$

for all $v\in\mathrm{H}^{\frac{d+1}{2}}(\mathbb{R}^d,\mathbb{R}^n)$. Here δ_x denotes the Dirac distribution supported at the point $x\in\mathbb{R}^d$. If $\alpha=0$ it holds that

$$(1-\Delta)^{\frac{d+1}{4}}u_{\mathbb{D},0}=\sum_{i=1}^m\lambda_i\delta_{x^i}, \tag{2.3}$$

in the weak sense for some $\lambda_i\in\mathbb{R}^n$ for $i=1,\dots,m$.

Proof Notice that, thanks to (0.1), it holds that $\delta_x\in\mathrm{H}^{-\frac{d+1}{2}}(\mathbb{R}^d,\mathbb{R})$ for each $x\in\mathbb{R}^d$. Taking variations in direction of any function ve_k for $k=1,\dots,n$, where e_k is the kth basis element of $\mathbb{R}^n$, with arbitrary $v\in\mathrm{H}^{\frac{d+1}{2}}(\mathbb{R}^d)$ yields the equations

$$\begin{aligned}0=\frac{d}{ds}\Big|_{s=0}E_\alpha(u_{\mathbb{D}}+sve_k)&=\frac{\alpha}{c_d}\int_{\mathbb{R}^d}\big[(1-\Delta)^{\frac{d+1}{4}}u_{\mathbb{D},k}\big](x)\big[(1-\Delta)^{\frac{d+1}{4}}v\big](x)\,dx\\&\quad-\sum_{i=1}^m\big[y^i_k-u_{\mathbb{D},k}(x^i)\big]\cdot v(x^i)\text{ for }k=1,\dots,n,\end{aligned}$$

for $v\in\mathrm{H}^{\frac{d+1}{2}}(\mathbb{R}^d)$, and where $u_{\mathbb{D},k}=(u_{\mathbb{D}})_k$. The identities amount to the validity of the system (2.2). The equivalence of the latter with (2.1) follows from

$$\sum_{i=1}^m\big[y^i-u_{\mathbb{D}}(x^i)\big]\cdot v(x^i)=\Big\langle\sum_{i=1}^m\big[y^i-u_{\mathbb{D}}(x^i)\big]\delta_{x^i},v\Big\rangle,\ v\in\mathrm{H}^{\frac{d+1}{2}}(\mathbb{R}^d,\mathbb{R}^n),$$

and the fact that the (pseudo)differential operator $(1-\Delta)^{\frac{d+1}{4}}$ is self-adjoint along with the validity of

$$\int_{\mathbb{R}^d}\big[(1-\Delta)^{\frac{d+1}{4}}u\big](x)\cdot\big[(1-\Delta)^{\frac{d+1}{4}}v\big](x)\,dx$$

$$= \left\langle (1-\Delta)^{\frac{d+1}{4}} u, (1-\Delta)^{\frac{d+1}{4}} v \right\rangle = \left\langle (1-\Delta)^{\frac{d+1}{2}} u, v \right\rangle \text{ for } u, v \in \mathrm{H}^{\frac{d+1}{2}}(\mathbb{R}^d, \mathbb{R}^n),$$

where the latter duality pairing is between the space $\mathrm{H}^{\frac{d+1}{2}}(\mathbb{R}^d, \mathbb{R}^n)$ and its dual $\mathrm{H}^{-\frac{d+1}{2}}(\mathbb{R}^d, \mathbb{R}^n)$. In the case when $\alpha = 0$, one takes variations in direction of test C^∞ functions with $\varphi(\mathbb{X}) = \not\vdash$ to obtain that

$$(1-\Delta)^{\frac{d+1}{2}} u = 0 \in \mathbb{R}^d \setminus \mathbb{X},$$

in the sense of distributions. This means that

$$\operatorname{supp}\big((1-\Delta)^{\frac{d+1}{2}} u\big) \subset \mathbb{X}.$$

It is know that compactly supported distributions are of finite order and thus it must hold that $(1-\Delta)^{\frac{d+1}{2}} u$ is a finite linear combinations of δ_{x^i}, $i = 1, \dots, m$, and derivatives of them. Notice, however, that $\partial_l \delta_x \notin \mathrm{H}^{-\frac{d+1}{2}}(\mathbb{R}^d)$ for $l = 1, \dots, d$ and so no derivatives can be present in the linear combination. This yields the claim. □

Lemma 2.3 *Let $u_{\mathbb{D},\alpha}$ be the minimizer of E_α for the data set $\mathbb{D}$. Then $u_{\mathbb{D},\alpha} \in \mathrm{C}^\infty(\mathbb{R}^d \setminus \mathbb{X}, \mathbb{R}^n)$, regardless of the regularization parameter $\alpha \geq 0$.*

Proof This regularity will readily follow from the reprentation of the solution discussed next. □

The minimization of E_0 or a variety of similar functionals has long been recognized to provide an answer to the so-called universal approximation property in the context of learning. It indeed can be shown that

$$u_{\mathbb{D},0} \to f \text{ as } |\mathbb{X}| \to \infty,$$

if the values $\mathbb{Y} = \big\{y^i \,\big|\, i = 1, \dots, m\big\}$ are those $\mathbb{Y} = f(\mathbb{X})$ of a function f belonging to a variety of (suitably chosen) functions spaces, such as $\mathrm{C}_c(\mathbb{R}^d, \mathbb{R}^n)$ (compactly supported continuous functions) or $\mathrm{L}^p(\mathbb{R}^d, \mathbb{R}^n)$ for $p \in [1, \infty)$. The convergence takes place in the space's natural topology as $\mathbb{X}$ becomes a finer and finer, not necessarily regular, discrete grid that fills the whole domain. The curse of dimensionality, however, limits the applicability of this approximation procedure as the size of any finite "filling" grid is exponential in the ambient dimension.

A reason the above approach or, more in general, kernel based approximation or interpolation has found widespread use, is its ability to bridge the gap between finite and infinite dimensional spaces. This property amounts in the case at hand in the possibility of computing the continuous variable solution $u_{\mathbb{D},\alpha}$ by solving finite dimensional systems.

Theorem 2.4 *The minimizer $u_{\mathbb{D},\alpha}$, $\alpha > 0$, is completely determined by its values $u_{\mathbb{X},\alpha} = u_{\mathbb{D},\alpha}\big|_{\mathbb{X}} = \big(u_{\mathbb{D},\alpha}(x^i)\big)_{i=1,\dots,m}$, on the set of arguments $\mathbb{X} = \big\{x^i \,\big|\, i = 1, \dots, m\big\}$. The latter can be determined by solving the well-posed linear system*

$$\big(\alpha + M_{\mathbb{D}}\big) u_{\mathbb{X},\alpha} = M_{\mathbb{D}} \mathbb{Y}, \tag{2.4}$$

where the matrix $M_{\mathbb{D}} \in \mathbb{R}^{m \times m}$ is given by

$$M_{ij} = \exp\big(-2\pi |x^i - x^j|\big),\ i, j = 1 \dots, m.$$

It then holds that

$$u_{\mathbb{D},\alpha}(x) = \frac{1}{\alpha}\sum_{j=1}^{m}\big[y^j - u^j_{\mathbb{X},\alpha}\big]\exp\big(-2\pi|x-x^j|\big), \tag{2.5}$$

for any $x \in \mathbb{R}^d$. *For* $\alpha = 0$, *it holds that*

$$\Lambda = M_{\mathbb{D}}^{-1}\mathbb{Y}, \text{ i.e. that } \lambda_i = \sum_{j=1}^{m}\big[M_{\mathbb{D}}^{-1}\big]_{ij}y^j$$

Proof The Fourier transform of the Laplace kernel is known. Indeed

$$\mathcal{F}_d\big[\exp(-2\pi\varepsilon|\cdot|)\big](\xi) = \frac{c_d\varepsilon}{\big(\varepsilon^2+|\xi|^2\big)^{\frac{d+1}{2}}}, \tag{2.6}$$

where $c_d = \Gamma(d+1)/\pi^{\frac{d+1}{2}}$. For the purposes of this paper the parameter ε is set to be 1. The right-hand side of the above identity is the symbol of the pseudo-differential operator $c_d(1-\Delta)^{-\frac{d+1}{2}}$. Now, equation (2.1) is equivalent to

$$\alpha\, u = \sum_{j=1}^{m}\big[y^j - u(x^j)\big]c_d(1-\Delta)^{-\frac{d+1}{2}}\delta_{x^j} = \sum_{j=1}^{m}\big[y^j - u(x^j)\big]\exp\big(-2\pi|\cdot - x^j|\big),$$

where the second equality sign follows from (2.6) and well-known properties of the Fourier transform. It only remains to evaluate this identity on the arguments $\mathbb{X}$ to obtain the finite linear system. When $\alpha = 0$, representation (2.3) entails that

$$u_{\mathbb{D},0} = \sum_{j=1}^{m}\lambda_j\exp\big(-2\pi|\cdot - x^j|\big),$$

and therefore that

$$\mathbb{Y} = u_{\mathbb{D},0}(\mathbb{X}) = M_{\mathbb{D}}\Lambda,$$

as claimed. □

Proposition 2.5 *The matrix* M *is invertible and it holds that*

$$u_{\mathbb{X},\alpha} \to \mathbb{Y} \text{ as } \alpha \to 0.$$

Proof The fact that M is invertible is a consequence of the fact that $\exp(-2\pi|\cdot - \cdot|)$ is a positive kernel as follows from its Fourier transform and the well-known characterization of positivity. Continuity of the inversion function inv

$$\mathrm{inv} : \mathrm{GL}_m \to \mathrm{GL}_m, M \mapsto M^{-1},$$

then shows that

$$u_{\mathbb{X},\alpha} = (\alpha + M_{\mathbb{D}})^{-1}M_{\mathbb{D}} \to M_{\mathbb{D}}^{-1}M_{\mathbb{D}}\mathbb{Y} = \mathbb{Y} \text{ as } \alpha \to 0.$$

□

Remark 2.6 The proof of Lemma 2.3 is now obvious since the explicit representation of $u_{\mathbb{D},\alpha}$ reveals that singularities are only found on the set $\mathbb{X}$.

Remark 2.7 The convergence $u_{\mathbb{X},\alpha} \to u_{\mathbb{X},0}$ as $\alpha \to 0$ implies convergence $u_{\mathbb{D},\alpha}$ to $u_{\mathbb{D},0}$ in the topology of $\mathrm{H}^{\frac{d+1}{2}}(\mathbb{R}^d,\mathbb{R}^n)$ and hence, uniform convergence as well, thanks the embedding (0.1).

Remark 2.8 Since the continuous minimization problem (0.5) has a unique solution, the linear system (2.4) is assured to be solvable and, in fact, well conditioned also in parameter ranges of interest (but of course not in general). It also follows that its solution $u_{\mathbb{X}}$, as well as its extension $u_{\mathbb{D}}$ to $\mathbb{R}^d$, depend continuously on the data $\mathbb{D}$ since the forcing term in (2.1) depends continuously on $\mathbb{D}$. The latter follows from the linear dependence on $\mathbb{Y}$ and the fact that Dirac distributions depend continuously on the location of their support in the topology of $\mathrm{H}^{-\frac{d+1}{2}}(\mathbb{R}^d)$, again a known consequence of (0.1).

Remark 2.9 Depending on the data set $\mathbb{D}$, the values $u_{\mathbb{X}}$ will be close or not so close to the prescribed values $\{y^i \,|\, i = 1,\dots,m\}$. Thus, if $u_{\mathbb{D}}$ is considered an interpolant of the data, it will not be exact, but only approximately capture the data. In many applications, some of which are considered in next section, this is a small price to be paid for the gain in robustness that the approach guarantees.

Remark 2.10 Using directly that $\frac{1}{c_d}(1-\Delta)^{-\frac{d+1}{2}}u_{\mathbb{D}} = \sum_{i=1}^m \lambda_i \delta_{x^i}$ for some $\lambda \in \mathbb{R}^m$, it is possible to derive a system of equations for λ using that $u_{\mathbb{D}} = \sum_{i=1}^m \lambda_i \exp\bigl(-2\pi|x-x^i|\bigr)$. Indeed it must hold

$$\alpha \sum_{i=1}^m \lambda_i \delta_{x^i} = \sum_{i=1}^m \bigl(y^i - u_{\mathbb{D}}(x^i)\bigr)\delta_{x^i},$$

which yields the system

$$\bigl(\alpha + M\bigr)\lambda = \mathbb{Y}.$$

This shows that $u_{\mathbb{D}}(x^i) = y^i - \alpha\lambda_i$ and, in particular, reiterates the point about the convergence as $\alpha \to 0$. In practice, it is more convenient to work with this system in order to compute $u_{\mathbb{D}}$.

While the parameter $\alpha \geq 0$ plays an important role, it will be dropped from the notation from now on. The understanding is that its value can be inferred from the context and that, whatever its value is, it is kept fixed. In this paper we are particularly focussed on the case $n = 1$ and on the trivial value set $\mathbb{Y} = \mathbb{1}$, where $y^i = 1$ for $i = 1,\dots,m$.

Definition 2.11 If $\mathbb{Y} = \mathbb{1}$, we say that $u_{\mathbb{D}}$ is the (continuous) *signal* generated by the data $\mathbb{X}$. We sometimes denote it by $u_{\mathbb{X}}$ or $u_{\mathbb{X},\mathbb{1}}$.

The signal is the inexact interpolation of the characteristic function of the data set[1]. It can be strong, if $u_{\mathbb{X}}(x^i) \simeq 1$, $i = 1,\dots,m$. This is the case, as was observed above, when $\mathbb{X}$ is a fine and locally filling discretization of the ambient space, such as when approximating a set of positive measure by a set of discrete points. More often, however, the signal will be weak in the sense that $u_{\mathbb{X}}(x^i)$ is significantly less than 1 for $i = 1,\dots,m$. In this paper we contend that the usefulness of the signal $u_{\mathbb{X}}$ does not only stem from its approximation or interpolation properties, but also (and perhaps mainly) from the

[1] We observe, in particular, that, if the data set discretizes a set S of measure 0, then then its characteristic function χ_S is the trivial function and of not much use.

fact that most of its level sets are very smooth due to Proposition 2.3 and, in fact, deliver smooth manifold approximations of $\mathbb{X}$ that can effectively be employed as stable decision boundaries in supervised classification problems. Superlevel sets of the signal are reliable approximations with positive measure of any discrete data set that prove robust against noise. They, in a sense, connect the dots and capture the shape of the data. Thus the main philosophical difference between the traditional view point, that considers the data as the manifestation of a function that needs to be reconstructed, and the view point taken in this paper is that here the data set itself is approximated by the mostly smooth (super)level sets of a function that may not even fit the data well at all. We are indeed more interested in the level surfaces generated by the data's signal than we are in its values. It will be shown that data signals, whether they are strong or weak, can be succesfully exploited in this sense. The practical experiments run in the next section will make use the following proposition.

Proposition 2.12 *Let* $\mathbb{X}_0 = \mathbb{X}_1 \dot\cup \mathbb{X}_2 \dot\cup \cdots \dot\cup \mathbb{X}_N$ *(the notation means that the union is disjoint) be a given data set consisting of* $N \in \mathbb{N}$ *subsets (or classes). For*

$$\mathbb{D}_0 = \big\{(x^i, 1) \,\big|\, i = 1, \dots, |\mathbb{X}_0|\big\},$$
$$\mathbb{D}_l = \big\{(x, 1) \,\big|\, x \in \mathbb{X}_l\big\} \cup \Big\{(x, 0) \,\Big|\, x \in \bigcup_{k \neq l} \mathbb{X}_k\Big\},\ l = 1, \dots, N,$$

it holds that

$$u_{\mathbb{D}_0} = \sum_{l=1}^{N} u_{\mathbb{D}_l}.$$

Proof The signal $u_{\mathbb{D}_0}$ and the signals $u_{\mathbb{D}_1}, \dots, u_{\mathbb{D}_N}$ relative to $\mathbb{X}_0$, are solutions of the linear equations

$$A_d\, u + u\big|_{\mathbb{X}_0} \cdot \delta_{\mathbb{X}_0} = \mathbb{Y}_l \cdot \delta_{\mathbb{X}_0},\ l = 0, \dots, N,$$

where $A_d = \frac{\alpha}{c_d}(1 - \Delta)^{\frac{d+1}{2}}$ and where

$$u\big|_{\mathbb{X}_0} \cdot \delta_{\mathbb{X}_0} = \sum_{i=1}^{|\mathbb{X}_0|} u(x^i)\delta_{x^i},\ \mathbb{Y}_l \cdot \delta_{\mathbb{X}_0} = \sum_{i=1}^{|\mathbb{X}_0|} y_l^i \delta_{x^i}.$$

The claim therefore follows from the fact that

$$\sum_{l=1}^{N} \mathbb{Y}_l = \mathbb{Y}_0.$$

□

Remark 2.13 Notice that signals do not, however, behave additively, in the sense that

$$u_{\mathbb{X}_1} + u_{\mathbb{X}_2} \neq u_{\mathbb{X}_1 \cup \mathbb{X}_2},$$

in general, even when $\mathbb{X}_1 \cap \mathbb{X}_2 = \emptyset$

If one is given a labeled data set $\mathbb{X}_0$, where N is the number of labels and $\mathbb{X}_l$, $l = 1, \dots, N$ are the subsets consisting of the data corresponding to label l, i.e. $L(x) = l$ for $x \in \mathbb{X}_l$, then one obtains a classification algorithm by computing the relative signals $u_{\mathbb{X}_l}$ for $l = 1, \dots, N$

and assigning any unlabeled datum $z \in \mathbb{R}^d$ to the class that exhibits the strongest relative signal, that is,

$$L(z) = \text{argmax}_l \, u_{\mathbb{X}_l}(z). \tag{2.7}$$

Remark 2.14 An important aspect of the proposed approach (and of kernel based methods as well) is that the fundamental solution

$$G(x) = \exp\big(-2\pi\,|x|\big),\ x \in \mathbb{R}^d, \tag{2.8}$$

of the (pseudo)differential operator $\frac{1}{c_d}(1-\Delta)^{\frac{d+1}{2}}$ used to obtain the finite linear system is essentially dimension independent. It depends on it only through the Euclidean distance function, which is a minimal ingredient that can hardly be avoided. It would of course be extremly difficult to work directly with the (pseudo)differential operator or the energy functional in high dimension. An application to the well-known MNIST classification problem will be discussed in the next section using an approach based on the signal generated by the training data. In that case one has that $d = 784$.

Remark 2.15 Just as for exact interpolation and due to the fact there are no constraints like e.g. boundary conditions, the method is completely local. This means that an approximant can be computed based on a subset of the original data set that is confined or restricted to a subregion of interest. This feature will be exploited in the MNIST classification problem.

Remark 2.16 It is well-known that the parameter $\alpha > 0$ has a regularizing effect that can be used to deal with noisy data when performing interpolation. It turns out that it also helps smooth out the level sets of $u_{\mathbb{D}}$. This is also illustrated in the next section.

Remark 2.17 It is sometimes convenient to modify the decay rate of the exponential "basis" functions, especially if the data undergoes some initial normalization. This can be done without significant consequences other than a slight modification of the objective functional or, equivalently, of the corresponding differential operator. Indeed, for $\gamma > 0$ and using the well-known scaling and translation properties of the Fourier transform $\mathcal{F}_d$, it holds that

$$\begin{aligned}
\Big(\frac{1}{c_d}(1-\gamma^2\Delta)^{\frac{d+1}{2}}\Big)u\big(\frac{\cdot_x - y}{\gamma}\big) &= \frac{1}{c_d}\mathcal{F}_d^{-1}\big(1+\gamma^2|\cdot_\xi|^2\big)^{\frac{d+1}{2}}\mathcal{F}_d\Big(\tau_y\,\sigma_{1/\gamma}u\Big)\\
&= \frac{1}{c_d}\mathcal{F}_d^{-1}\Big(\big(1+|\gamma\cdot_\xi|^2\big)^{\frac{d+1}{2}}\exp\big(-i\frac{y}{\gamma}\cdot\gamma\cdot_\xi\big)\gamma^n\widehat{u}(\gamma\cdot_\xi)\Big)\\
&= \frac{1}{c_d}\mathcal{F}_d^{-1}\Big[\gamma^n\sigma_\gamma\Big(\exp\big(-i\frac{y}{\gamma}\cdot\cdot_\xi\big)\big(1+|\cdot_\xi|^2\big)^{\frac{d+1}{2}}\widehat{u}\Big)\Big]\\
&= \sigma_{1/\gamma}\Big(\frac{1}{c_d}(1-\Delta)^{\frac{d+1}{2}}u\Big)\big(\cdot_x - \frac{y}{\gamma}\big)\\
&= \Big(\frac{1}{c_d}(1-\Delta)^{\frac{d+1}{2}}u\Big)\big(\frac{\cdot_x - y}{\gamma}\big).
\end{aligned}$$

Here we use the notation $\widehat{u} = \mathcal{F}_d(u)$ for the Fourier transform of the function u as well as $\cdot_x$ and $\cdot_\xi$ as place holders for the independent variables x and ξ in order to distinguish a function from its values. Furthermore τ_y denotes translation, that is, $\big(\tau_y u\big)(\cdot) = u(\cdot - y)$,

and σ_γ scaling, or $\big(\sigma_\gamma u\big)(\cdot) = u(\gamma\cdot)$. Replacing u by $\exp\big(-2\pi|\cdot|\big)$ and $y = x^i$ for any $i = 1, \dots, m$ it is seen that

$$\frac{1}{c_d}(1 - 4\pi^2\Delta)^{\frac{d+1}{2}} \exp\big(-|\cdot - x_i|\big) = \delta_{x^i}.$$

Thus the use of the modified exponentials merely corresponds to an inconsequential modification of the Euler-Lagrange equation (or its generating functional).

Related approaches

In this section we highlight some important connections between the point of view just described and well established frameworks.

Reproducing kernel Hilbert spaces (RKHS)

When $\alpha = 0$, it can be seen that $\mathrm{H}^{\frac{d+1}{2}}$ is a RKHS with kernel K given by the fundamental solution G of $(1 - \Delta)^{\frac{d+1}{2}}$ via $K(x, y) = G(x - y)$. Indeed, it holds by construction that

$$\begin{aligned}\big(K(\cdot, y)\big|u\big)_{\mathrm{H}^{\frac{d+1}{2}}} &= \big((1 - \Delta)^{\frac{d+1}{4}} K(\cdot, y)\big|(1 - \Delta)^{\frac{d+1}{4}} u\big)_{\mathrm{L}^2} \\ &= \big\langle(1 - \Delta)^{\frac{d+1}{2}} K(\cdot, y), u\big\rangle = \langle\delta_y, u\rangle = u(y),\ \forall\, u \in \mathrm{H}^{\frac{d+1}{2}}.\end{aligned}$$

As $\delta_{x_1}, \dots, \delta_{x_m}$ are linearly independent vectors when the points are distinct and $(1 - \Delta)^{\frac{d+1}{2}}$ is an isomorphism between $\mathrm{H}^{\frac{d+1}{2}}$ and $\mathrm{H}^{-\frac{d+1}{2}}$, the functions $K(\cdot, x_1), \dots, K(\cdot, x_m)$ are linearly independent and thus $\mathrm{H}^{\frac{d+1}{2}}$ is a fully interpolating RKHS according to the definition given in [7]. It follows, in particular, that equation (2.4) is the equation that determines the minimal norm interpolant for the data $(\mathbb{X}, \mathbb{Y})$. In applications it is more common to start with a positive definite kernel (see [7] for a definition). While this may make little difference from a purely pragmatic point of view, it somewhat obfuscates the exact nature of the corresponding RKHS space and its norm. It is our contention that the norm can be chosen in a way as to shape the features of the associated kernel and interpolation. The discriminating power of the method arguably owes more to the choice of norm (and, hence, kernel) than to any adjustable parameter that may be present in the kernel function. The commonly used polynomial, exponential, and sigmoid kernels are all smooth in contrast to the kernel chosen here which has a carefully chosen regularity that determines the properties of the corresponding interpolant. The RKHS space is chosen as large as possible compatible with the continuity requirement discussed earlier.

Kernel support vector machines (K-SVM)

There is also a connection to SVM that use the kernel trick in order to maintain the idea of linear separation but to apply it to a feature vector generated by a special nonlinear transformation of the data set $\mathbb{X}$. As the method reduces to computations involving only the scalar product of feature vectors, that can be easily computed via a kernel, this procedure does not render the method impractical (the feature vectors typically live in a possibly much higher dimension than the original data). In this paper, the motivation is rather to allow all nonlinear functions of a large function space to compete in order to interpolate the "data manifold" via their level sets. The choice of space is made in order to ensure the possibility of sharp (but continuous) transitions while maintaining the smoothness of (almost all) level sets. Notice that the choice of kernel in K-SVM is a straighforward

trick that, however, conceals potential difficulties due to the increased dimensionality not necessarily present in the original data. We refer to [8, Section 12.3.4] for a more detailed discussion and only point out here that a choice of kernel may not necessarily be aligned with the structure of the data and make it hard for the method to identify the "data manifold" in the presence of noise.

Ridge regression

Another related method admitting solutions that can be described by kernels is that of ridge regression. The connection appears when $\alpha > 0$. In kernel ridge regression [8] a data interpolant is looked for in the form

$$u(x) = \lambda \cdot h(x) + \lambda_0,$$

where $\lambda \in \mathbb{R}^n$ and $h = (h_1, \ldots, h_n)$ is a vector of basis functions, by minimizing a measure of the error with a regularizing term given as a multiple of $|\lambda|^2$. In that case, if one defines the kernel $K_n(x, y) = \sum_{j=1}^{n} h_j(x)h_j(y)$, a solution can be found in the form $u = \sum_{i=1}^{m} \lambda_i K(\cdot, x_i)$ leading to a system very much like (2.4). Formally letting n tend to ∞, one would obtain a problem in infinite dimensions if the functions h_j were chosen to be the eigenfunctions of some kernel (where available). The penalizing norm would, however, converge to an L^2-type norm [2] and, moreover, a discrete set of eigenfunctions may not be available in general, as it is the case for the Laplace kernel used here. It is interesting to observe that the method described in this paper, like RKHS methods ($\alpha = 0$), finds an infinite dimensional problem that necessarily has a solution which can be obtained by solving a finite dimensional one.

In the general category of data-smoothing models one can also find those developed and studied by Wahba and her school, see [9] for a nice account, where, starting with the one dimensional case, the functional

$$\frac{1}{m}\sum_{i=1}^{m}\left(y^i - u(x^i)\right)^2 + \lambda \int_a^b \left|u_{xx}\right|^2 dx$$

is used to generate piecewise cubic polynomials as interpolants on an interval $[a, b] \subset \mathbb{R}$. The method was also extended to include other interpolants by using other differential operators and made practical for moderate dimensions by a clever (and not immediately obvious) use of tensor products. The approach taken in this paper is similar in spirit but is truly ambient dimension independent.

Numerical experiments

In this section a series of experiments are performed to illustrate the effectiveness of the method described in the previous section. First we consider two dimensional problems to highlight important aspects and in order to motivate and justify the use of the method in a high dimensional context. The section then concludes with an application to the classification of the MNIST data set.

Stability of signals' level sets

Working in the context of approximating measurable functions, simple functions play an important role as they are the building block of any measurable function. While the

[2] unless one uses weighted norms.

approximation property is well-known we present a few examples to illustrate the efficacy of the use of the data signal's level sets for classification purposes. First we will consider situations where the approximation is good, then examples when it is rather poor. It will be shown that, in all cases, i.e., regardless of how good the approximation is, the signal's level sets still provide very useful information. This observation is crucial since it opens the door to applications to high dimensional data, where it is inconceivable that the data arguments $\mathbb{X}$ provide a fine grid of even a small portion of the ambient space.

Characteristic functions of sets of positive measure

Take the three subsets of $\mathbb{R}^2$

$$S_1 = \{x \in \mathbb{R}^2 \,|\, |x| \leq .6\}, \tag{4.1}$$

$$S_2 = \{x \in \mathbb{R}^2 \,|\, |x_1| + |x_2| \leq .7\}, \tag{4.2}$$

$$S_3 = \{r(\theta)(\cos(\theta), \sin(\theta)) \in \mathbb{R}^2 \,|\, .4 \leq r(\theta) \leq .6 + .1\cos(4\theta),\ \theta \in [0, 2\pi)\}, \tag{4.3}$$

and consider the associated characteristic function χ_{S_j} for $j = 1, 2, 3$. The first data set consists of the values of these functions on a regular grid, that is,

$$\mathbb{D}_{m,j} = \{(x^i, \chi_{S_j}(x^i)) \,|\, i = 1, \ldots, m^2\},\ j = 1, 2, 3,$$

for $\mathbb{X}_m = \{(kh - 1, lh - 1) \,|\, 0 \leq k, l \leq m - 1\}$ and $h = 2/(m - 1)$, which amounts to a uniform discretization of the box $B = [-1, 1] \times [-1, 1]$. In Fig. 1 some level lines of the interpolant $u_{\mathbb{D}_{m,j}}$ are shown for the three functions $\chi_{S_j}, j = 1, 2, 3$, and for different values of the regularizing parameter $\alpha > 0$ and $m = 16$. While the size of the data set clearly correlates with the "accuracy" of the interpolation, the approximating function does an excellent job at generating meaningful and smooth level sets. Their smoothness is affected mainly by the parameter α and the their proximity to the level sets corresponding to the highest values.

It follows that if a characteristic function has to be recovered or inferred from a data set, thresholding based on the interpolant $u_{\mathbb{D}_{m,j}}$ is an effective strategy and the decision boundary $[u_{\mathbb{D}_{m,j}} = .5\max(u_{\mathbb{D}_{m,j}})]$ is a solid choice across a range of values of the regularization parameter. Figure 2 depicts the same experiments using the denser data sets $\mathbb{D}_{32,j}$ for $j = 1, 2, 3$.

In Figs. 3 and 4, it is shown how the method performs in the presence of data corruption. In Fig. 3 2% of the data is misclassified, whereas the misclassification rate in Fig. 4 is 5%. By this we mean that a mistake is made, with the given probability, when a value is assigned to an argument by evaluating the corresponding characteristic function. These examples clearly demonstrate the usefulness of the regularizing parameter which leads to data signals whose decision level sets are more stable in the presence of classification errors.

Sample data

Finally we demonstrate that the method offers a degree robustness when the data arguments are randomly sampled from a uniform distribution supported on the box B. The resulting decision boundaries of half maximal value are depicted in Fig. 5 along with the sampled argument data sets $\mathbb{X}_m$. The sampling rate clearly affects the smoothness of the level sets, a deterioration that is to some extent counteracted by the regularization.

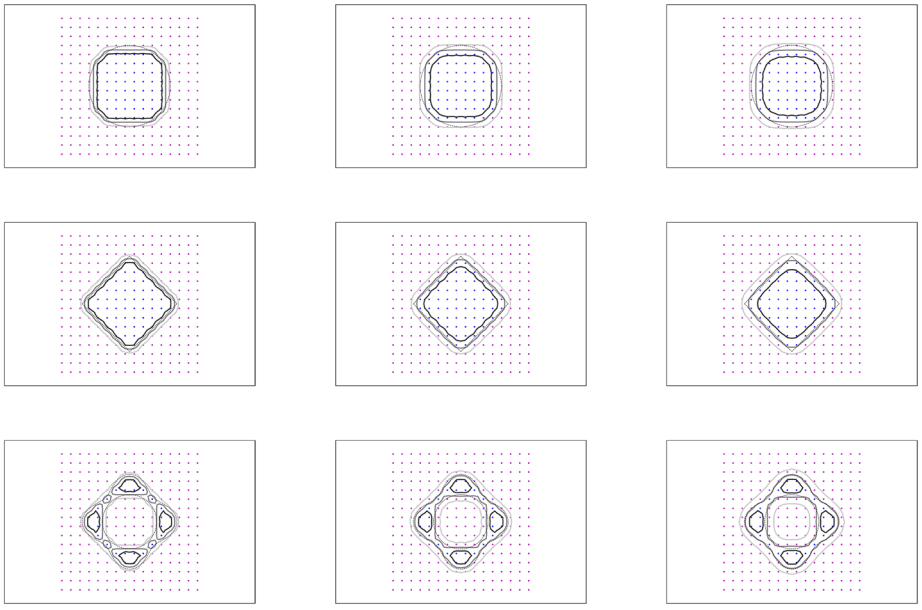

Fig. 1 Level lines of the signal $u_{\mathbb{D}_{mj}}$ for $m = 16$, $j = 1, 2, 3$, and $\boldsymbol{\alpha} = 0.1, 1.0, 2.0$. Depicted are the data set (blue dots correspond to the value 1 while magenta dots to the value 0) and three level lines corresponding to levels at 20%, 50%, and 80% of the signal's maximal value, respectively. Darker lines correspond to higher levels. The parameter $\boldsymbol{\alpha}$ grows from left to right. The boundary of the set S_j appears as a dashed black line

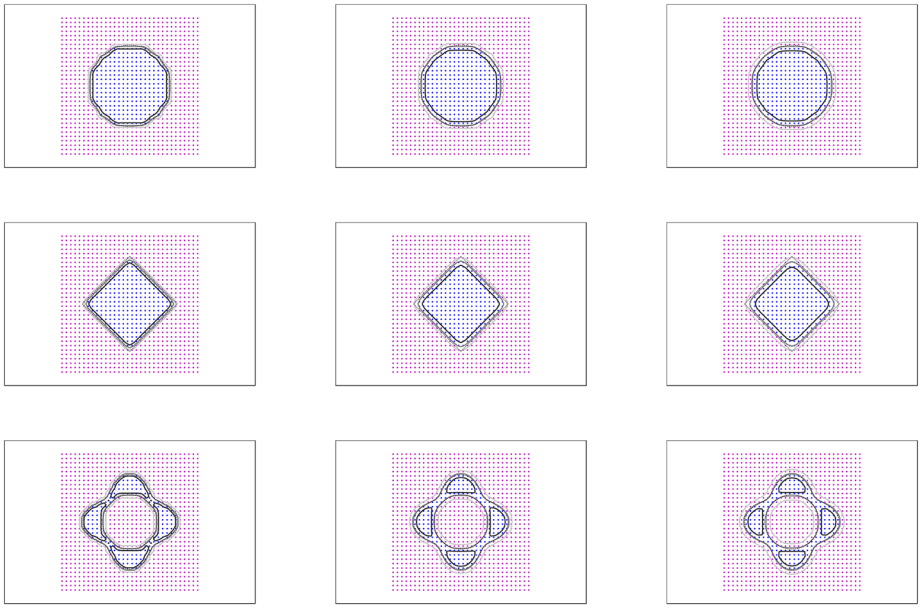

Fig. 2 Level lines of the signal $u_{\mathbb{D}_{mj}}$ for $m = 32$, $j = 1, 2, 3$, and $\boldsymbol{\alpha} = 0, 1, 1, 2$. Depicted are the data set (blue dots correspond to the value 1 while magenta dots to the value 0) and three level lines corresponding to levels at 20%, 50%, and 80% of the signal's maximal value, respectively. Darker lines correspond to higher levels. The parameter $\boldsymbol{\alpha}$ grows from left to right. The boundary of the set S_j appears as a dashed black line

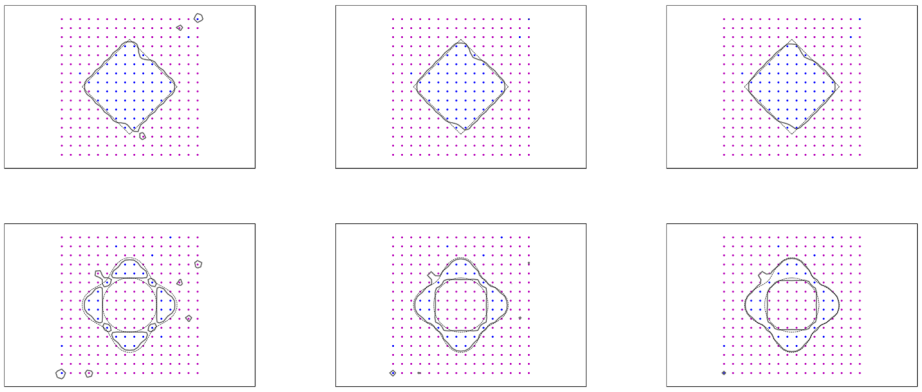

Fig. 3 Decision boundary of the signal $u_{\mathbb{D}_{mj}}$ for $m = 16, j = 2, 3$, and $a = 0, 1, 1, 2$ with 2% data corruption rate. Depicted are the data set (blue dots correspond to the value 1 while magenta dots to the value 0) and the level line corresponding to 50% of the signal's maximal value. The parameter $\boldsymbol{\alpha}$ grows from left to right. The boundary of the set S_j appears as a dashed black line

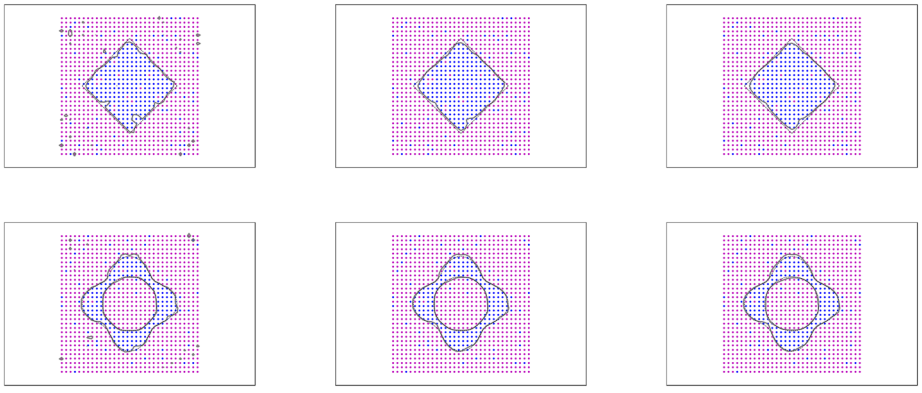

Fig. 4 Decision boundary of the signal $u_{\mathbb{D}_{mj}}$ for $m = 32, j = 2, 3$, and $a = 0, 1, 1, 2$ with 5% data corruption rate. Depicted are the data set (blue dots correspond to the value 1 while magenta dots to the value 0) and the level line corresponding to 50% of the signal's maximal value. The parameter $\boldsymbol{\alpha}$ grows from left to right. The boundary of the set S_j appears as a dashed black line

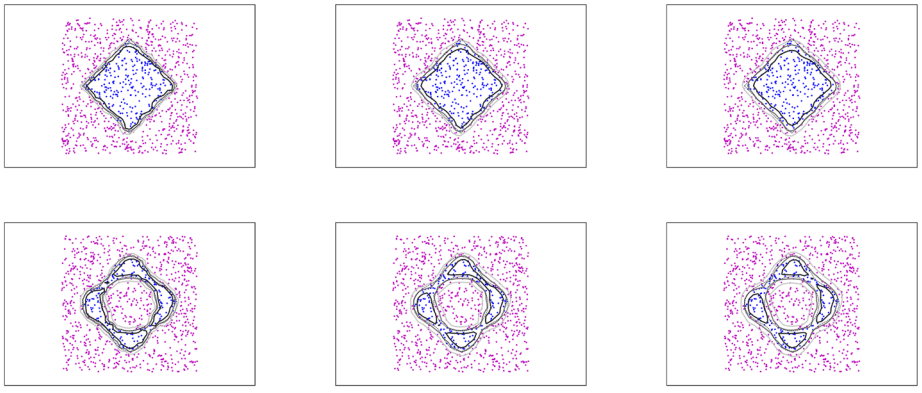

Fig. 5 Level lines of 20%, 50%, and 80%, increasingly dark, of the signal's maximal value for regularizations parameter $\boldsymbol{\alpha} = .1, 1, 2.$, from left to right. The number of randomly sampled points, also depicted with the associated value color-coded (with blue representing the value 1, while magenta the value 0), is 1024 as in the denser regular grids of previous examples

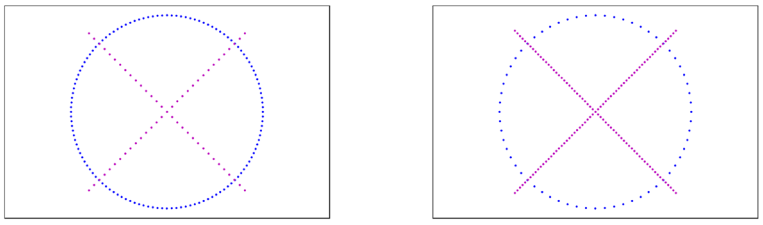

Fig. 6 From left to right, the data set pairs (C_k, S_k), $k = 1, 2$. They can be thought as different samplings of the same pair of "continuous" sets

Classification

Continuity of the interpolant and its level sets (almost all of them actually smooth) are obtained at the cost of approximate interpolation. Such an approximation can still be accurate when the argument data set covers the function's domain of definition uniformly and the value set is accurate, but the real advantage of this method is its applicability to incomplete data and/or noisy data sets. This point is further reinforced with the next series of experiments, where the data build a lower dimensional manifold of the ambient space and its signal is weak, that is, information about the underlying function is limited to sets of zero measure, or the data is not deterministic (in its argument set) but only has a probability distribution for its location.

Consider the data sets $\mathbb{D}_k, k = 1, 2$, consisting of points belonging to two distinct classes: points along a circle and points on the union of two segments with two different densities as depicted in Fig. 6. For $k = 1, 2$, denote the circular data sets by

$$C_k = \left\{c_k^i \,\middle|\, i = 1, \dots m_{C_k}\right\}$$

and the union of segments data sets by

$$S_k = \left\{s_k^i \,\middle|\, i = 1, \dots m_{S_k}\right\}$$

In the spirit of the previous examples, we create two pairs of data sets

$$\begin{aligned}\mathbb{D}_{C_k} &= \left\{(c_k^i, 1) \,\middle|\, i = 1, \dots m_{C_k}\right\} \cup \left\{(s_k^i, 0) \,\middle|\, i = 1, \dots m_{S_k}\right\} \text{ and}\\ \mathbb{D}_{S_k} &= \left\{(s_k^i, 1) \,\middle|\, i = 1, \dots m_{S_k}\right\} \cup \left\{(c_k^i, 0) \,\middle|\, i = 1, \dots m_{C_k}\right\},\ k = 1, 2\end{aligned}$$

and compute the associated signals $u_{\mathbb{D}_{C_k}}$ and $u_{\mathbb{D}_{S_k}}$, $k = 1, 2$. Figure 7 shows the level sets

$$\left[u_{\mathbb{D}_{C_k}} = .5 \max(u_{\mathbb{D}_{C_k}})\right] \text{ and } \left[u_{\mathbb{D}_{S_k}} = .5 \max(u_{\mathbb{D}_{S_k}})\right],\ k = 1, 2.$$

for different values of the regularization parameter.

The region in their interior (i.e. the one containing the corresponding data set) can be considered as a smooth fattening of the data set to a set of positive measure. It can be obtained for any data set regardless of the intensity of its signal.

Next we turn our interest to the question of classification: given a point $z \in \mathbb{R}^2$ that needs to be classified, we use the decision algorithm defined by (2.7). This gives

$$L(z) = \text{argmax}_{l=1,2}\, u_{\mathbb{D}_l}(z),$$

which, in this case, yields the level lines (hypersurfaces in higher dimension)

$$\left[u_{\mathbb{D}_1} = u_{\mathbb{D}_2}\right] \tag{4.4}$$

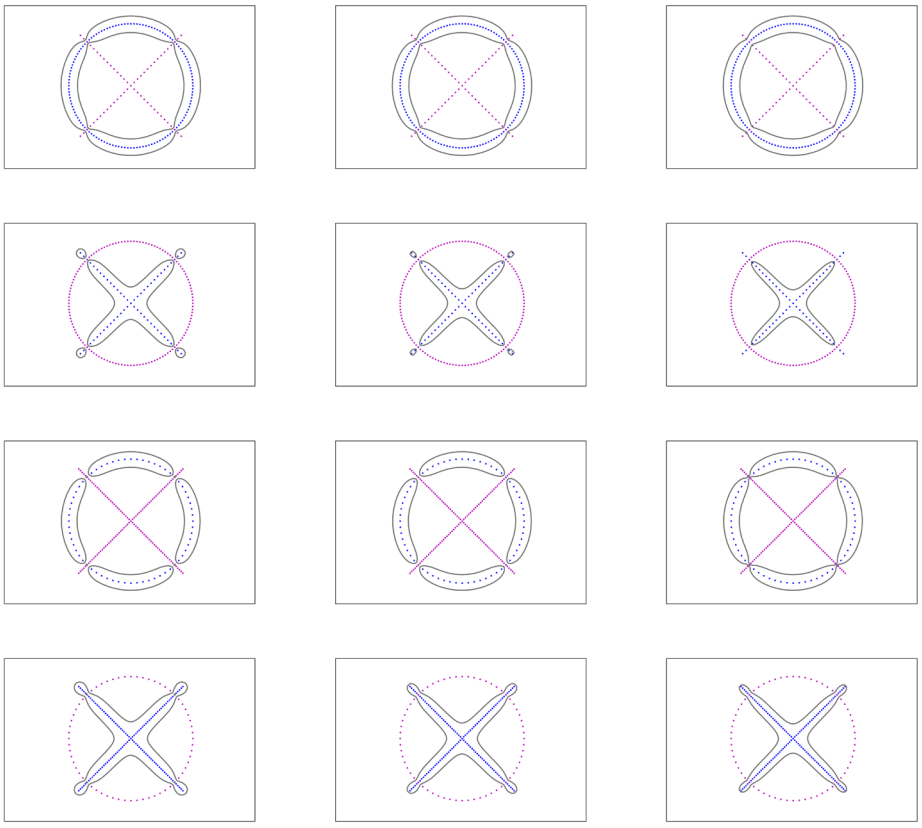

Fig. 7 The 50% of maximum level lines for the signals $u_{\mathbb{D}_C}$ and $u_{\mathbb{D}_S}$ of the data sets based on the two data classes C_k and S_k for $k = 1, 2$. The first two rows correspond to $k = 1$ and the second two to $k = 2$. Withing each row, from left to right, the regularization parameter is $\alpha = .1, 1, 2$

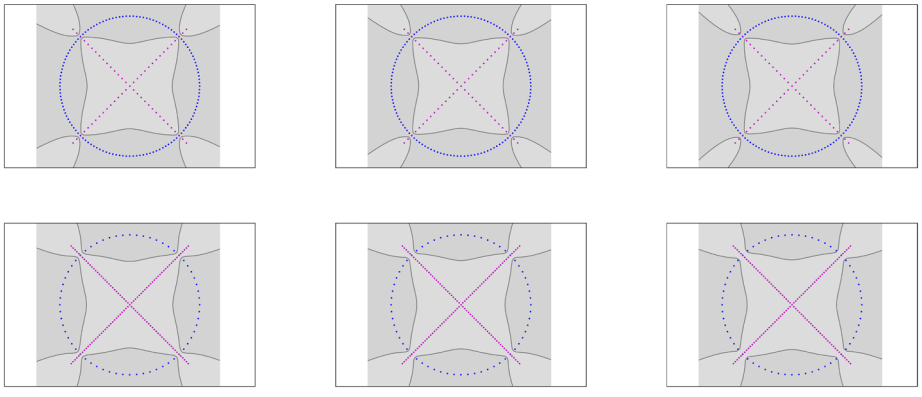

Fig. 8 The decision boundary computed according to (4.4) for two classes depicted in the same image. The kth row corresponds to the class pair (C_k, S_k), $k = 1, 2$. Notice how the decision boundary is affected by the "density" of the data sets and not very strongly affected by the regularization parameter. The latter is, from left to right, $\alpha = .1, 1, 2$

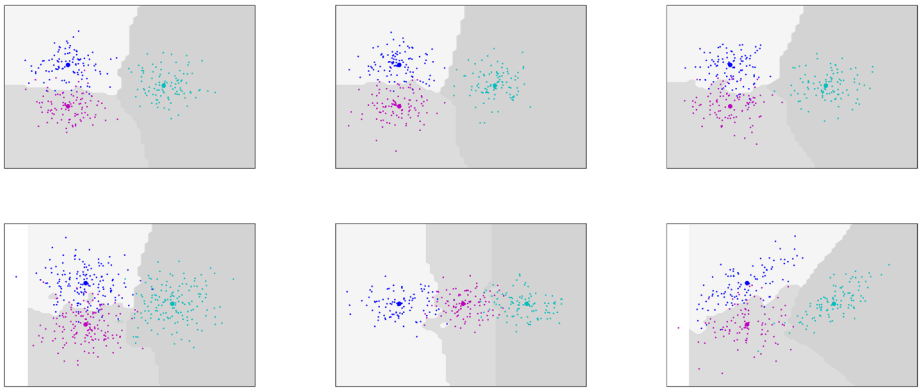

Fig. 9 The decision regions computed according to (2.7) for three classes of normally distributed points depicted using different colors in the same image. The average of each class is plotted as a large disk. The first row shows three different realization of the same three normal distributions and the computed decision regions. The first image in the second row is based on the same distributions as the first row but the sample size is increased, the second and third show the decision regions for different choices of means and covariance matrices. In all but the last example the covariance is taken to be diagonal. In the third and the last image, one of the sample points is outside the computational box where the data signals are generated and hence generated an unshaded region

as the decision boundary. This is illustrated in Fig. 8 for the classification problem of the two class pairs (C_k, S_k) for $k = 1, 2$ introduced above.

If the data pairs (C_k, S_k), $k = 1, 2$, are considered the ground truth, then the above decision boundary is arguably optimal. If, on the other hand, it is known that the actual sets are the continuous circle and the union of two segments, then the data are only a sampling of these sets. In this case, the decision boundary may be biased by the relative oversampling of the one set compared to the other. This is evident when one compares the decision boundaries of Fig. 8. In concrete situations, if information about the dimensionality of the ground truth is known, this effect can be mitigated by using comparable sampling rates for the different classes (see next section for an example of this procedure) or by normalizing the data fidelity term to read $\frac{1}{2m}\sum_{i=1}^{m}\left|u(x^i) - y^i\right|^2$.

We conclude this section with a classification problem for data $\mathbb{X}_0$ split into three classes $\mathbb{X}_l$, $l = 1, 2, 3$, each consisting of a set of points which are normally distributed with mean $p_l \in \mathbb{R}^2$ and different covariance matrices. The data set and the corresponding decision regions computed based on (2.7) are depicted in Fig. 9. In these experiments $\alpha = 1$.

The MNIST data set

The final application is to the standard machine learning example and toy problem of digit classification for the MNIST data set.

Remark 2.18 The purpose of this example is to illustrate how the choice of regularizer allows for the method to be used for data that live in high dimension. There is no effort to optimize the parameter choices nor to compete with state-of-the-art algorithms. The idea is rather to show how a theoretically justified and derived method with a small number of parameters can perform an acceptable job on a high dimensional problem. The method has its limitations as it assumes that the data classes (almost) lie on separable manifolds. This may be approximately true for MNIST but is certainly not true for other data sets.

The data set consists of 28×28 grayscale images of hand-written digits stored as vectors in $[0, 255]^{784}$. Here it is considered that the ambient space is simply $\mathbb{R}^{784}$. The argument data is normalized to have unit Euclidean norm, that is, each original vector x is replaced by $x/|x|$. In this way the maximal Euclidean distance between any two data points is $2\sqrt{2}$. The data set is split into a training set containing 60,000 data points x and their corresponding label $d(x)$ indicating which digit is represented, and a testing data set of size 10,000. The labels of the testing data are known but need to be inferred from any knowledge that can be gleaned from the training set. This an example of when, due to the so-called curse of dimensionality, the data does not have any hope to fill the ambient space uniformly and thus, even if one assumed the existence of an underlying function $d : \mathbb{R}^{784} \to \{0, 1, \ldots, 9\}$, the data would never be sufficient to accurately approximate it. It has to be said of course, that the testing data mostly does not stray away significantly from the training data and the different digits in the latter build thin subsets of the ambient space. This fact is typically captured by saying that the data lives in some lower dimensional manifold(s). We know from the previous section and from the two dimensional experiments, however, that the (training) data still generates a significant, if not strong, signal. The classification method described in the sequel exploits this signal and does not require any kind of training based on the minimization of non-convex functionals, as is often the case for machine learning algorithms based on neural nets. It is, in fact, based on the solution of low dimensional, well-posed linear systems as is about to be explained. First, in order to strengthen the signal somewhat, the training set is expanded to include rotations by $\pm 10°$ and horizontal/vertical translations by ± 2 pixels of each image. Then, given a test image z, the closest 5 training images of each digit class are determined

$$\mathbb{X} = \left\{ x^{i_j} \,\middle|\, j = 1, \ldots, 50 \right\},$$

where $d(x^{i_j}) = \lfloor j/5 \rfloor$. The idea is now to use system (2.4) in order to produce approximate interpolants $u_d := u_{\mathbb{D}_d}$ of the characteristic functions of each digit class given by the data set

$$\mathbb{D}_d = \left\{ \left(x^{i_j}, \delta_{d,d(x^{i_j})} \right) \,\middle|\, j = 1, \ldots, 50 \right\},$$

where $d = 0, \ldots, 9$, and

$$\delta_{d,\bar{d}} = \begin{cases} 1, & d = \bar{d}, \\ 0, & d \neq \bar{d}. \end{cases}$$

Finally the approximative characteristic functions u_d will compete to determine the digit $d(z)$ to be associated with the test image z via (2.7), in this case

$$d(z) = \operatorname{argmax}_d u_d(z).$$

This approach yields an accuracy of 98.56%[3] on the test set. In Table 1 we record the detailed outcome of the classification, performed with $\alpha = 1.5$.

Recall that this method is stable and depends continuously on the data set and hence delivers a robustness that methods with higher classification rates typically do not. Moreover, unlike neural networks, it does not require any training but uses the training set directly in a fully transparent way.

[3] For comparison, a classification based on a direct nearest neighbor approach using the extended training set has an accuracy of 97.86%

Table 1 MNIST classification results: the kth row of the table indicates in column j how many times the digit k is assigned the label j by the algorithm

Label										
Digit	0	1	2	3	4	5	6	7	8	9
0	973	0	1	0	0	2	3	1	0	0
1	0	1131	2	1	0	0	0	1	0	0
2	4	1	1015	0	1	0	0	10	0	1
3	0	0	1	995	0	8	0	3	3	0
4	0	0	0	0	971	0	4	1	0	6
5	1	0	0	8	0	877	4	1	0	1
6	0	2	0	0	0	1	955	0	0	0
7	0	4	4	0	0	0	0	1020	0	0
8	2	0	3	9	3	2	3	4	946	2
9	1	2	1	4	9	7	0	10	2	973

Remark 2.19 It should be pointed out that the proposed classification method performs well when the data classes effectively lie on submanifolds the shape of which their relative signals are able to capture. If the class similarity is not geometric in this sense, this method will likely not produce satisfactory results if applied to the original data. It is indeed possible for general data sets to exhbit classes that share some common feature but are far apart as points in space. In this case, the dots cannot be easily connected.

Acknowledgements
None.

Author contributions
The author read and approved the final manuscript.

Funding
None.

Availability of data and material
Upon request.

Declarations

Competing interests
The authors declare that they have no competing interests.

Received: 7 June 2023 Accepted: 12 December 2023
Published online: 03 January 2024

References

1. Wendland H. Scattered data approximations. Cambridge monographs on applied and computational mathematics. Cambridge: Cambridge University Press; 2004.
2. Vapnik VN, Chervonenkis AY. On a class of algorithms of learning pattern recognition. Avtomatika i Telemekhanika. 1964;25(6):In Russian.
3. Aizerman MA, Braverman EM, Rozonoer LI. Theoretical foundations of the potential function method in pattern recognition learning. Autom Remote Control. 1964;25:821–37.
4. Boser BE, Guyon IM, Vapnik VN. A training algorithm for optimal margin classifiers. In Proceedings of the fifth annual Acm workshop on Computational learning theory, pages 144. Assn for Computing Machinery. 1992.
5. Poggio T, Girosi F. Regularization algorithms for learning that are equivalent to multilayer networks. Science. 1990;247(4945):978–82.
6. Bousquet O, Elisseeff A, Stability and generalization. J Mach Learn Res. 2002;2.
7. Paulsen VI, Raghupathi M. An introduction to the theory of reproducing kernel Hilbert spaces. Cambridge Studies in Advanced Mathematics 152. Cambridge University Press, Cambridge. 2016.
8. Hastie T, Tibshirani R, Friedman J. The elements of statistical learning. Data mining, inference, and prediction. Springer Series in Statistics. Springer, New York. 2009.
9. Gu C, Wang Y. Grace Wahba and the Wisconsin spline school. Notices of the American Mathematical Society. 2022;69(3).