Kernapproximation mit Quasi-Monte-Carlo-Methoden und Anwendung auf die Kern-Ridge-Regression

Gloria Maria Hellner

Geboren am 4. April 2003 in Lennestadt

17. Februar 2025

Bachelorarbeit Mathematik

Betreuer: Prof. Dr. Jochen Garcke

Zweitgutachter: Prof. Dr. Jürgen Dölz

Institut für Numerische Simulation

Mathematisch-Naturwissenschaftliche Fakultät der

Rheinischen Friedrich-Wilhelms-Universität Bonn

Inhaltsverzeichnis

1. Einleitung

In der heutigen Zeit spielen maschinelle Lernverfahren eine zentrale Rolle in zahlreichen Anwendungsbereichen, von der Finanzwelt bis hin zur medizinischen Diagnostik. Ziel des maschinellen Lernens ist es, aus Daten Strukturen zu erkennen, um Klassifikations- und Regressionsaufgaben zu lösen.

Die Bewertung von Immobilien ist eine zentrale Herausforderung in Wirtschaft und Finanzwesen. Der Preis eines Objekts hängt von zahlreichen Faktoren wie Lage, Größe und Ausstattung ab. Allerdings sind diese Zusammenhänge oft nichtlinear. Kleine Unterschiede in der Umgebung können erhebliche Preisunterschiede bewirken. Dies macht die präzise Modellierung solcher Abhängigkeiten zu einer anspruchsvollen Aufgabe, für die klassische lineare Regressionsmethoden oft nicht ausreichen (vgl. [Pre+25]).

Ein Beispiel aus der medizinischen Diagnostik ist die Klassifikation von Brustkrebs (vgl. [Seo+22]). Brustkrebs ist eine der am häufigsten diagnostizierten Krebsarten bei Frauen. Da eine frühzeitige Diagnose unerlässlich ist, um das Fortschreiten der Erkrankung zu verhindern, wurden in den letzten Jahren zahlreiche maschinelle Lernmodelle entwickelt, die die histopathologische Klassifikation der verschiedenen Karzinomtypen automatisieren sollen. Allerdings sind viele dieser Ansätze nicht auf großskalige Datensätze skalierbar.

Eine wesentliche Herausforderung im maschinellen Lernen besteht darin, unbekannte Datenpunkte auf Grundlage bereits vorhandener Trainingsdaten vorherzusagen. Ein leistungsfähiges Werkzeug für diese Art von Problemen sind Kernmethoden, die nichtlineare Beziehungen zwischen den Daten erfassen können (vgl. [SS02]). Ein klassischer Ansatz zur Vorhersage mit Kernmethoden ist die Kern-Ridge-Regression (KRR), die Regularisierung einsetzt, um Überanpassung zu vermeiden. Ein entscheidender Nachteil der klassischen KRR ist jedoch der hohe Speicherbedarf von $\mathcal{O}(N^2)$ und die Rechenkomplexität von $\mathcal{O}(N^3)$, was die Methode für große Datensätze unpraktisch macht.

Zur Verbesserung der Skalierbarkeit wurde der Ansatz der Approximation der Kernfunktion mittels Random Features (RF) entwickelt (vgl. [RR17]). Diese Methode basiert auf der Monte-Carlo-Methode (MC-Methode) und nutzt die Darstellung der Kernfunktion als Integral. Indem die Approximation der Kernfunktion als Skalarprodukt dargestellt wird, lassen sich die Berechnungen effizienter durchführen. Jedoch bringt dieser Ansatz auch Nachteile mit sich, insbesondere aufgrund der probabilistischen Konvergenzordnung von $\mathcal{O}\left(\frac{1}{\sqrt{M}}\right)$. Eine vielversprechende Alternative stellt die Quasi-Monte-Carlo-Methode (QMC-Methode) dar, bei der anstelle von zufälligen Features deterministische Punktmengen mit niedriger Diskrepanz verwendet werden, um die Konvergenzordnung zu verbessern (vgl. [Avr+16], [HSH24]). Ein Problem dieser Methode ist, dass der Integrand in einigen Fällen eine unendliche Variation aufweisen kann, was die Anwendung klassischer Fehlerschranken nicht möglich macht (vgl. [Avr+16]). Dennoch zeigt [HSH24], dass es möglich ist, Fehlerschranken der Ordnung $\mathcal{O}\left(\frac{1}{M}\right)$, bis auf logarithmische Faktoren, zu formulieren.

Das Ziel dieser Arbeit ist es, die theoretischen Ergebnisse dieser Methoden zu untersuchen und die numerischen Experimente aus [HSH24] zu reproduzieren und zu erweitern. Dabei werden die Genauigkeit und Effizienz der exakten KRR sowie der beiden Approximationsansätze verglichen, um die theoretischen Ergebnisse zu verifizieren.

Übersicht

Zunächst führen wir in Kapitel 2 die Grundlagen der Kernmethoden sowie die KRR ein. Im Anschluss widmen wir uns dem RF-Ansatz zur Approximation von Kernfunktionen. Hierbei diskutieren wir die Integraldarstellung von Kernen, die MC-Methode und deren Anwendung auf die Integraldarstellung (Kapitel 3 und 4). Anschließend stellen wir in Kapitel 5 die QMC-Methode als Alternative vor. Wir erläutern Beispiele für Folgen mit niedriger Diskrepanz und analysieren den Approximationsfehler des Integrals. In Kapitel 6 wenden wir die QMC-Methode auf die Integraldarstellung des Kerns an. Dabei unterscheiden wir zwischen translationsinvarianten und nicht-translationsinvarianten Kernen, formulieren in beiden Fällen geeignete Bedingungen und leiten Fehlerschranken für die Kernapproximation ab.

Im Kapitel 7 wenden wir sowohl den RF-Ansatz als auch den QMC-Ansatz auf die KRR an. Dabei führen wir die Approximation des Integraloperators und der Kernmatrix ein, um theoretische Ergebnisse für die Quasi-Monte-Carlo-Kern-Ridge-Regression (QMC-KRR) zu präsentieren und diese mit den Resultaten der Random-Feature-Kern-Ridge-Regression (RF-KRR) zu vergleichen.

Zum Abschluss validieren wir unsere theoretischen Erkenntnisse durch numerische Experimente (Kapitel 8).

Eigener Beitrag

- Aufarbeitung der Grundlagen der KRR und der Kernapproximation mittels MC- und QMC-Methoden.
- Aufarbeitung theoretischer Ergebnisse aus [HSH24] für die QMC-KRR.
- Eigenständige Implementierung der KRR und deren Approximationen via RF und der QMC-Methode.
- Reproduktion der in [HSH24] beschriebenen numerischen Experimente anhand synthetischer Daten zum Vergleich der beiden Approximationsansätze sowie zusätzliche Tests für alternative Regressionsfunktionen und in höheren Dimensionen.
- Implementierung weiterer numerischer Experimente: die Einbeziehung alternativer QMC-Folgen, Vergleich der Laufzeiten zwischen exaktem Algorithmus und Approximationen sowie Variation des Parameters σ.
- Anwendung der Implementierung auf einen realen Datensatz mit und ohne vorherige Dimensionsreduktion.

2. Kernmethoden

2.1 Grundlagen der Kernmethoden

Kernmethoden sind ein leistungsstarkes Werkzeug im maschinellen Lernen. Sie ermöglichen es, komplexe Probleme in Bereichen wie Klassifikation, Regression und Dimensionsreduktion zu lösen. Anwendungen finden Kernmethoden unter anderem in der Support Vector Machine (SVM) für Klassifikationsaufgaben (vgl. [SC08]), der Gauß-Prozess-Regression (GPR) (vgl. [RW06]) sowie der Kern-Ridge-Regression (KRR) für regularisierte Regressionen (vgl. [SS02]) und der Kern-PCA für Dimensionsreduktion (vgl. [SSM98]). Insbesondere ermöglichen sie die Anwendung linearer Methoden auf nichtlineare Probleme durch eine geeignete Transformation der Daten in höherdimensionale Merkmalsräume.

In diesem Kapitel werden die mathematischen Grundlagen der Kernmethoden erläutert, die als Grundlage für die späteren Kapitel dienen. Die Inhalte dieses Unterkapitels basieren auf [SS02].

Um Probleme wie Klassifikation und Regression im maschinellen Lernen zu lösen, benötigen wir ein Ähnlichkeitsmaß. Wir verfügen über Daten, die in einer Menge X enthalten sind und wollen wissen, wann zwei Datenpunkte $x, x' \in X$ als ähnlich gelten. An X bestehen keine besonderen Anforderungen, es muss lediglich eine Menge sein. Um die Ähnlichkeit zu messen, transformieren wir die Datenpunkte aus X in einen Raum, in dem ein Skalarprodukt definiert ist:

$$\phi : X \to \mathcal{F}.$$

Den Raum $\mathcal{F}$ nennen wir Merkmalsraum (Feature Space) und die Abbildung ϕ Merkmalsabbildung (Feature Map).

Damit können wir ein Ähnlichkeitsmaß definieren:

$$k(x, x') := \langle \phi(x), \phi(x') \rangle_{\mathcal{F}}.$$

Als Nächstes definieren wir den Begriff des symmetrischen Kerns und veranschaulichen ihn anhand klassischer Beispiele.

Definition 2.1 (symmetrischer Kern)**.** Sei $X \subset \mathbb{R}^d$. Eine Abbildung $k : X \times X \to \mathbb{R}$ heißt Kern. Wir nennen den Kern symmetrisch, falls $k(x, x') = k(x', x)$ für alle $x, x' \in X$ gilt.

Beispiele: Sei $X = \mathbb{R}^d$. Häufig genutzte Kerne sind

- Linearer Kern: $k(x, x') = x^\intercal x'$.
- Polynom-Kern: $k(x, x') = (x^\intercal x' + c)^d$, wobei $c \geq 0$.
- Min-Kern: $k(x, x') = \prod_{i=1}^{d} \min\{x_i, x'_i\}$.
- Gauß-Kern: $k(x, x') = \exp(-\frac{\|x - x'\|_2^2}{2\sigma^2})$, wobei $\sigma \in \mathbb{R}$.

Kernmethoden bieten viele Vorteile:

1. **Nichtlinearität**: Kerne ermöglichen die Modellierung nichtlinearer Beziehungen in den Daten, indem sie die Daten in einen Raum transformieren, in dem lineare Methoden angewendet werden können.
2. **Effizienz**: Die Berechnung von Skalarprodukten im Merkmalsraum kann oft effizient durch die Kernfunktion erfolgen, ohne die explizite Berechnung der Merkmalsabbildung (Kern-Trick).
3. **Flexibilität**: Es gibt eine Vielzahl von Kernen (z. B. Polynom-Kerne, Gauß-Kern), die für unterschiedliche Problemstellungen geeignet sind.

Ein grundlegendes Konzept bei der Anwendung von Kernmethoden ist die Eigenschaft der positiven Semidefinitheit.

Definition 2.2 (symmetrischer und positiv semidefiniter Kern)**.** Ein Kern $k : X \times X \to \mathbb{R}$ heißt symmetrisch und positiv semidefinit (spsd), wenn für alle $X_n = \{x_1, \ldots, x_n\} \subset X$ gilt, dass die Kernmatrix $K = [k(x_i, x_j)]_{i,j=1}^n$ symmetrisch und positiv semidefinit ist.

Im Folgenden betrachten wir ausschließlich symmetrische und positiv semidefinite Kerne, sofern nicht anders angegeben.

Um die Methoden in einem geeigneten Funktionsraum durchzuführen, konstruieren wir einen Hilbertraum. Dazu betrachten wir einen spsd Kern k auf einer Menge X. Wir definieren einen Funktionsraum durch

$$\operatorname{span}\{k(x_i, \cdot) | x_i \in X\}.$$

Auf diesem Raum können wir ein Skalarprodukt definieren. Für $f = \sum_{i=1}^n \alpha_i k(x_i, \cdot)$ und $g = \sum_{i=1}^{n'} \beta_i k(x'_i, \cdot)$ definieren wir

$$\langle f, g \rangle := \sum_{i=1}^{n} \sum_{j=1}^{n'} \alpha_i \beta_j k(x_i, x'_j).$$

Es lässt sich zeigen, dass dies ein wohldefiniertes Skalarprodukt ist (vgl. [SS02]).

Indem wir den Abschluss dieses Funktionsraumes bilden, erhalten wir einen Hilbertraum.

Definition 2.3 (Hilbertraum mit reproduzierendem Kern (RKHS, engl. Reproducing Kernel Hilbert Space))**.** Ein Hilbertraum $\mathcal{H}$ von Funktionen auf einer Menge X mit Skalarprodukt $\langle \cdot, \cdot \rangle_{\mathcal{H}}$ heißt Hilbertraum mit reproduzierendem Kern, wenn eine Kernfunktion $k : X \times X \to \mathbb{R}$ existiert, sodass $k(x, \cdot) \in \mathcal{H}$ für alle $x \in X$ und

$$f(x) = \langle f, k(x, \cdot) \rangle_{\mathcal{H}} \quad \text{für alle } x \in X \text{ und } f \in \mathcal{H}.$$

Der Kern k heißt reproduzierender Kern.

In diesem Fall gilt

$$\mathcal{H} = \overline{\operatorname{span}\{k(x, \cdot) | x \in X\}}.$$

Der durch diesen Hilbertraum induzierte reproduzierende Kern ist eindeutig. Diese Räume spielen eine zentrale Rolle für Kernmethoden, da sie den theoretischen Rahmen bieten, in dem Kernfunktionen sinnvoll interpretiert und angewendet werden können. Insbesondere

ermöglicht die Konstruktion eines RKHS, dass Funktionen in einem strukturierten und vollständigen Raum mit einem wohldefinierten Skalarprodukt operieren.

Zusätzlich zu den oben erläuterten Eigenschaften induzieren Kerne Integraloperatoren, die eine zentrale Rolle bei Kernmethoden spielen.

Definition 2.4 (Integraloperator)**.** Sei $k : X \times X \to \mathbb{R}$ ein stetiger Kern, X kompakt und sei $\boldsymbol{X}$ eine Zufallsvariable mit Werten in X und der Verteilung $P_{\boldsymbol{X}}$. Dann ist der Integraloperator des Kerns k definiert als

$$L : L^2(P_{\boldsymbol{X}}) \to L^2(P_{\boldsymbol{X}})$$

$$\begin{aligned} Lf(x) &:= \mathbb{E}_{\boldsymbol{X} \sim P_{\boldsymbol{X}}}[k(\boldsymbol{X}, x) f(\boldsymbol{X})] \\ &= \int_X k(z, x) f(z) \, \mathrm{d}P_{\boldsymbol{X}}(z). \end{aligned} \tag{2.1}$$

Dabei bezeichnet $L^2(P_{\boldsymbol{X}})$ den Raum der quadratintegrierbaren Funktionen auf X bzgl. $P_{\boldsymbol{X}}$.

Wir definieren das Bild des Integraloperators als

$$\operatorname{ran} L := \{Lf \mid f \in L^2(P_{\boldsymbol{X}})\}$$

(vgl. [HSH24], A.3.A).

Die in diesem Kapitel behandelten Konzepte legen die theoretische Grundlage für die folgenden Kapitel, in denen wir kernbasierte Vorhersageverfahren und die KRR detailliert betrachten.

2.2 Kernbasierte Vorhersageverfahren

Die Inhalte dieses Unterkapitels basieren auf [SS02], Kapitel 3 und 4.

In vielen maschinellen Lernverfahren steht man vor der zentralen Herausforderung, Modelle zu entwickeln, die sowohl präzise Vorhersagen treffen als auch robust gegenüber Störungen in den Daten bleiben. Insbesondere bei der Analyse realer Datensätze muss berücksichtigt werden, dass Beobachtungen grundsätzlich mit Rauschen behaftet sind, das heißt, es gilt $y_i = f(x_i) + \epsilon$ mit einem Fehlerterm ϵ. Eine exakte Interpolation der Trainingsdaten wäre hier kontraproduktiv, da sie Rauschen übernimmt (Überanpassung) und numerisch instabil wird (schlecht konditionierte Kernmatrizen). Ziel ist daher, eine Funktion $f \in \mathcal{H}$ zu finden, die die zugrundeliegende Datenstruktur modelliert und gleichzeitig durch Regularisierung stabil bleibt.

Um den beschriebenen Zielkonflikt zwischen einer guten Anpassung an die Daten und der Vermeidung numerischer Instabilität mathematisch zu erfassen, wird der Begriff der Verlustfunktion eingeführt. Diese misst, wie stark die Vorhersagen einer Funktion f von den beobachteten Zielwerten abweichen und bildet die Grundlage für eine stabilisierte Optimierung.

Definition 2.5 (Verlustfunktion)**.** Sei (X, Σ) ein messbarer Raum und $Y \subset \mathbb{R}$ eine abgeschlossene Teilmenge. Eine Funktion $\ell : X \times Y \times \mathbb{R} \to [0, \infty)$ heißt Verlustfunktion, wenn sie messbar ist und für alle $x \in X$ und $y \in Y$ die Eigenschaft $\ell(x, y, y) = 0$ erfüllt.

Definition 2.6 (Empirisches Risiko)**.** Für Daten $\{(x_i, y_i)\}_{i=1}^N \subset X \times Y$ ist das empirische Risiko einer Funktion $f : X \to \mathbb{R}$ definiert als

$$\mathcal{R}_{\ell,emp}(f) = \frac{1}{N} \sum_{i=1}^{N} \ell(x_i, y_i, f(x_i)).$$

Das empirische Risiko misst die durchschnittliche Abweichung der Vorhersagen $f(x_i)$ von den Zielwerten y_i.

Da die Minimierung des empirischen Risikos alleine oft zu Überanpassung oder numerischen Problemen führt, fügen wir einen Regularisierungsterm $\lambda \cdot s(\|f\|_{\mathcal{H}})$ hinzu, um die Stabilität der Lösung zu gewährleisten. Damit erhalten wir das regularisierte Risiko, welches wir minimieren möchten.

Satz 2.7 (Representer Theorem)**.** *Sei $s : [0, \infty) \to \mathbb{R}$ eine streng monoton wachsende Funktion, $\lambda > 0$, X eine Menge, $\mathcal{H}$ ein RKHS über X mit reproduzierendem Kern k und sei $\ell : X \times Y \times \mathbb{R}$ eine Verlustfunktion. Für eine gegebene Datenmenge $D := \{(x_i, y_i)\}_{i=1}^N \subset X \times Y$ existiert ein Minimierer $f \in \mathcal{H}$ des regularisierten Risikos*

$$\mathcal{R}_{\ell,reg}(f) = \frac{1}{N} \sum_{i=1}^{N} \ell(x_i, y_i, f(x_i)) + \lambda s(\|f\|_{\mathcal{H}}) \tag{2.2}$$

der Form

$$f(x) = \sum_{i=1}^{N} \alpha_i k(x_i, x).$$

Der Satz zeigt, dass die optimale Vorhersagefunktion eine Linearkombination der Kernauswertungen an den Trainingspunkten ist. Dadurch wird das Optimierungsproblem im potenziell unendlichdimensionalen RKHS auf die Bestimmung von N Koeffizienten α_i reduziert. Dies ermöglicht die effiziente Lösung durch lineare Gleichungssysteme.

2.3 Kern-Ridge-Regression

Die Inhalte dieses Unterkapitels basieren auf [SS02; FHH17].

Die Kern-Ridge-Regression (KRR) ist ein regularisierter Ansatz zur nichtlinearen Funktionsapproximation, der Überanpassung vermeidet und numerisch stabile Lösungen liefert. Dazu betrachten wir die quadratische Verlustfunktion

$$\ell(x_i, y_i, f(x_i)) = (f(x_i) - y_i)^2$$

und den Regularisierungsterm $s(\|f\|_{\mathcal{H}}) = \|f\|_{\mathcal{H}}^2$. Das Ziel ist es, für gegebene Daten $D = \{(x_i, y_i)\}_{i=1}^N$ das folgende Optimierungsproblem zu lösen:

$$\min_{f \in \mathcal{H}} \frac{1}{N} \sum_{i=1}^{N} (f(x_i) - y_i)^2 + \lambda \|f\|_{\mathcal{H}}^2. \tag{2.3}$$

Dabei ist $\lambda > 0$ der Regularisierungsparameter, der den Kompromiss zwischen Anpassung und Glattheit der Lösung steuert. Gemäß Satz 2.7 ist die Lösung dieses Problems durch die Darstellung

$$f(x) = \sum_{i=1}^{N} \alpha_i k(x_i, x). \tag{2.4}$$

gegeben, wobei k der reproduzierende Kern ist und $\alpha = (\alpha_1, \ldots, \alpha_N)^\intercal$ die zu bestimmenden Koeffizienten sind. Durch Einsetzen von (2.4) in (2.3) ergibt sich:

$$\frac{1}{N}\sum_{i=1}^{N}\left(\sum_{j=1}^{N}\alpha_j k(x_j, x_i) - y_i\right)^2 + \lambda \sum_{i,j=1}^{N}\alpha_i\alpha_j k(x_i, x_j).$$

Die Ableitung nach α_k und Nullsetzen führt für alle $k \in \{1, \ldots, N\}$ zu:

$$\frac{2}{N}\sum_{i=1}^{N}\left(\sum_{j=1}^{N}\alpha_j k(x_j, x_i) - y_i\right)k(x_k, x_i) + 2\lambda\sum_{i=1}^{N}\alpha_i k(x_k, x_i) = 0.$$

In Matrixschreibweise lässt sich dies zusammenfassen als:

$$\begin{aligned}\frac{2}{N}K(K\alpha - y) + 2\lambda K\alpha &= 0\\ K(K + \lambda N I_N)\alpha &= Ky\\ (K + \lambda N I_N)\alpha &= y\end{aligned}$$

wobei $K = [k(x_i, x_j)]_{i,j=1}^N$ die Kernmatrix ist und $y = [y_i]_{i=1}^N$.

Damit ist die Lösung von (2.3) von der Form (2.4), wobei α durch die Lösung des linearen Gleichungssystems

$$(K + N\lambda I_N)\alpha = y \tag{2.5}$$

gegeben ist.

Der Rechenaufwand der exakten KRR beträgt $\mathcal{O}(N^3)$, da das Gleichungssystem eine Matrixinversion der Größe $N \times N$ erfordert. Der Speicherbedarf beträgt $\mathcal{O}(N^2)$ aufgrund der Speicherung der Kernmatrix K.

Dieser Aufwand wird bei großen Datensätzen problematisch. Daher streben wir eine Verbesserung an, indem wir die Kernfunktion approximieren. Dadurch können wir die Berechnungen deutlich beschleunigen, wie wir in den folgenden Kapiteln zeigen wollen.

3. Integraldarstellung von Kernen

Die Inhalte dieses Kapitels basieren auf [HSH24; Avr+16].

Integraldarstellungen sind besonders vorteilhaft, da Integrale mit verschiedenen Methoden effizient approximiert werden können. Ein natürlicher Schritt ist daher, diese Idee auf die Kernfunktion anzuwenden, indem wir sie als Integral darstellen. Diese Darstellung eröffnet die Möglichkeit, die Kernfunktion durch geeignete Approximationen zu vereinfachen und so die Berechnungen zu optimieren.

Sei $k : X \times X \to \mathbb{R}$ ein Kern. Das Ziel besteht darin, k in der Form

$$k(x, x') = \int_\Omega \psi(x, \omega)\psi(x', \omega) \, \mathrm{d}\pi(\omega) \tag{3.1}$$

darzustellen, wobei π ein Wahrscheinlichkeitsmaß auf Ω und $\psi : X \times \Omega \to \mathbb{R}$ eine geeignete Funktion ist. Diese Darstellung bildet die Grundlage für Random Features und Quasi-Monte-Carlo-Approximationen der Kernfunktion (vgl. Kapitel 4 und 5). Die folgende Proposition liefert die Bedingungen für die Existenz einer solchen Integraldarstellung.

Proposition 3.1 (vgl. [HSH24])**.** *Sei $k : X \times X \to \mathbb{R}$ ein stetiger Kern, μ ein Maß mit vollem Träger auf X und es gelte*

$$\int_X k(x, x) \, \mathrm{d}\mu(x) < \infty.$$

Dann existiert eine Funktion $\psi : X \times \Omega \to \mathbb{R}$, sodass

$$k(x, x') = \int_\Omega \psi(x, \omega)\psi(x', \omega) \, \mathrm{d}\mu(\omega).$$

Eine zentrale Klasse von Kernen sind die translationsinvarianten Kerne. Ihre wesentliche Eigenschaft besteht darin, dass der Wert der Kernfunktion nicht von den absoluten Positionen der Eingabewerte abhängt, sondern ausschließlich von deren Differenz.

Definition 3.2 (vgl. [Avr+16])**.** Ein Kern $k : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{C}$ heißt translationsinvariant, wenn eine positiv definite Funktion $h : \mathbb{R}^d \to \mathbb{C}$ existiert, sodass $k(x, x') = h(x - x')$ für alle $x, x' \in \mathbb{R}^d$.

Für diese Kerne können wir die Integraldarstellung mithilfe des folgenden Satzes angeben.

Satz 3.3 (vgl. [Boc33])**.** *Eine Funktion $h : \mathbb{R}^d \to \mathbb{C}$ ist genau dann positiv definit, wenn sie die Fourier-Transformation eines endlichen nicht-negativen Borel-Maßes μ auf $\mathbb{R}^d$ ist, das heißt*

$$h(x) = \int_{\mathbb{R}^d} e^{-ix^\intercal \omega} \, \mathrm{d}\mu(\omega) \qquad \forall x \in \mathbb{R}^d.$$

Dieser Satz impliziert, dass translationsinvariante Kerne durch ein Wahrscheinlichkeitsmaß μ dargestellt werden können, sodass

$$\begin{aligned} k(x, x') = h(x - x') &= \int_{\mathbb{R}^d} e^{-i(x-x')^\intercal \omega} \, \mathrm{d}\mu(\omega) \\ &= \int_{\mathbb{R}^d} \int_0^{2\pi} \frac{1}{\pi} \cos(x^\intercal \omega + b) \cos((x')^\intercal \omega + b) \, \mathrm{d}b \, \mathrm{d}\mu(\omega). \end{aligned}$$

Diese Darstellung erlaubt es, konkrete Wahrscheinlichkeitsmaße μ für verschiedene Kernfunktionen zu spezifizieren. Im Folgenden sind einige häufig verwendete translationsinvariante Kerne und ihre zugehörigen Wahrscheinlichkeitsmaße aufgeführt.

Beispiele: (vgl. [HSH24])

- Gauß-Kern $e^{-\frac{\|\sigma(x-x')\|_2^2}{2}}$: $\mu \sim \mathcal{N}(0, \sigma^2 I_d)$.
- Laplace-Kern $e^{-\|\gamma(x-x')\|_1}$: μ hat die Lebesgue-Dichte $\prod_{i=1}^{d} \frac{1}{\pi\gamma(1+(\omega_i/\gamma)^2)}$ (Cauchy-Verteilung).
- Cauchy-Kern $\prod_{i=1}^{d} \frac{1}{1+(x_i-x_i')^2/\lambda^2}$: μ hat die Lebesgue-Dichte $\frac{\lambda}{2} e^{-\lambda\|\omega\|_1}$ (Laplace-Verteilung).

Angenommen, μ ist ein Wahrscheinlichkeitsmaß mit unabhängigen Komponenten, wobei die i-te Komponente die Verteilungsfunktion ϕ_i, $i = 1, \ldots, d$ besitzt. Definiere $\Phi(t) := (\phi_1(t_1), \ldots, \phi_d(t_d))^\intercal$ und $\Phi^{-1}(t) := (\phi_1^{-1}(t_1), \ldots, \phi_d^{-1}(t_d))^\intercal$ wobei ϕ_i^{-1} die Inverse der Verteilungsfunktion ist. Mit diesen Definitionen lässt sich der Kern k wie folgt schreiben:

$$\begin{aligned} k(x, x') &= h(x - x') \\ &= \int_{[0,1]^{d+1}} 2\cos(x^\intercal \Phi^{-1}(t) + 2\pi b)\cos((x')^\intercal \Phi^{-1}(t) + 2\pi b)\,\mathrm{d}b\,\mathrm{d}t. \end{aligned} \tag{3.2}$$

Damit erhalten wir die Darstellung des Kerns in der Form von Gleichung (3.1), wobei $\omega = (t, b) \in \mathbb{R}^d \times \mathbb{R}$ gleichverteilt auf $[0,1]^{d+1}$ ist und $\psi(x, \omega) = \sqrt{2}\cos(x^\intercal \Phi^{-1}(t) + 2\pi b)$. Dies ist vorteilhaft, da wir dank der Transformation Φ^{-1} nicht mehr die unterschiedlichen Verteilungen der Komponenten berücksichtigen müssen, sondern stattdessen die Einfachheit der Gleichverteilung ausnutzen können.

4. Kernapproximation mit Random Features

Die Approximation von Kernfunktionen verbessert die Effizienz von Kernmethoden erheblich. Insbesondere ermöglicht die Integraldarstellung eines Kerns $k(x, x')$, wie in (3.1) beschrieben, eine numerische Approximation. Random Features (vgl. [RR07]) nutzen die Monte-Carlo-Methode (MC-Methode) zur Approximation dieser Integrale, wodurch Kernberechnungen auf lineare Operationen reduziert und Speicher- sowie Rechenaufwand verringert werden. Dieses Kapitel behandelt zunächst die theoretischen Grundlagen der MC-Methode und anschließend deren Anwendung zur Approximation von Kernfunktionen.

4.1 Die Monte-Carlo-Methode

Die MC-Methode bietet eine effektive Möglichkeit, Integrale zu approximieren. Im Folgenden fassen wir die wesentlichen Grundlagen zusammen. Die Inhalte dieses Unterkapitels basieren auf [Nie92], Kapitel 1.2.

Ziel ist es, das Integral

$$\int_B f(x)\,\mathrm{d}x$$

zu approximieren, wobei $B \subset \mathbb{R}^d$, $0 < \lambda(B) < \infty$ und λ das d-dimensionale Lebesgue-Maß ist. Wir betrachten B als einen Wahrscheinlichkeitsraum mit Maß $\mathrm{d}\mu = \frac{\mathrm{d}x}{\lambda(B)}$. Dann gilt für $f \in L^1(\mu)$

$$\int_B f(x)\,\mathrm{d}x = \lambda(B) \int_B f\,\mathrm{d}\mu = \lambda(B)\mathbb{E}[f]$$

wobei $\mathbb{E}[f]$ der Erwartungswert der Zufallsvariable f ist. Damit reduziert sich die Approximation eines Integrals auf die Schätzung eines Erwartungswerts.

Sei nun f eine Zufallsvariable auf einem beliebigen Wahrscheinlichkeitsraum $(A, \mathcal{A}, \lambda)$. Der Monte-Carlo-Schätzer für den Erwartungswert $\mathbb{E}[f]$ ergibt sich durch M unabhängige, λ-verteilte zufällige Stichproben $a_1, \ldots, a_M \in A$ und wird wie folgt berechnet:

$$\mathbb{E}[f] \approx \frac{1}{M} \sum_{i=1}^{M} f(a_i). \tag{4.1}$$

Das Gesetz der großen Zahlen impliziert

$$\lim_{M\to\infty} \frac{1}{M} \sum_{i=1}^{M} f(a_i) = \mathbb{E}[f] \qquad \lambda^\infty\text{-f.ü.}$$

wobei λ^∞ das Produktmaß ist von abzählbar vielen Kopien von λ. Mit zunehmender Anzahl M von Stichproben konvergiert der Schätzer gegen den tatsächlichen Erwartungswert.

Der Fehler, der bei der Approximation in (4.1) entsteht, beträgt im Durchschnitt $\frac{\sigma(f)}{\sqrt{M}}$, wobei $\sigma(f)$ die Standardabweichung der Zufallsvariable f ist.

Obwohl die Methode einfach anzuwenden ist, hat sie einige Nachteile:

1. Die Fehlerschranke ist rein probabilistisch und keine deterministische Größe.
2. Die Regularität des Integranden f wird nicht berücksichtigt, was die Effizienz der Methode begrenzt.
3. Das Generieren qualitativ hochwertiger Zufallsstichproben kann in der Praxis herausfordernd sein.

Ebenso motiviert der vergleichsweise langsame Fehlerabbau von $\mathcal{O}\left(\frac{1}{\sqrt{M}}\right)$ die Suche nach effizienteren Alternativen. In Kapitel 5 werden wir die Quasi-Monte-Carlo-Methode als mögliche Verbesserung einführen.

4.2 Random Features

Die Inhalte dieses Unterkapitels basieren auf [RR07; HSH24].

Random Features wurden von Rahimi und Recht in [RR07] eingeführt, um eine effiziente Approximation von Kernmethoden zu ermöglichen. Dabei wird das Integral in (3.2) mithilfe der MC-Methode approximiert, indem $\{\omega_i\}_{i=1}^M$ unabhängig und identisch verteilt aus $\mathrm{Unif}[0,1]^{d+1}$ gewählt werden. Dies führt zur Approximation $k(x,x') \approx k_M(x,x')$ mit

$$k_M(x,x') = \frac{1}{M}\sum_{i=1}^{M} 2\cos(x^\intercal \Phi^{-1}(t_i) + 2\pi b_i)\cos((x')^\intercal \Phi^{-1}(t_i) + 2\pi b_i), \tag{4.2}$$

wobei $\omega_i = (t_i, b_i)$.

Alternativ kann das Integral aus (3.1) direkt mittels der MC-Methode approximiert werden, indem die $\{\omega_i\}_{i=1}^M$ unabhängig und identisch verteilt aus π gewählt werden. Dadurch ergibt sich die Approximation

$$k_M(x,x') = \frac{1}{M}\sum_{i=1}^{M} \psi(x,\omega_i)\psi(x',\omega_i). \tag{4.3}$$

Der Fehler dieser Approximation ist probabilistisch von der Ordnung $\mathcal{O}\left(\frac{1}{\sqrt{M}}\right)$.

Im Gegensatz zum klassischen Kern-Trick, bei dem der Wert der Kernfunktion direkt berechnet wird, ohne explizit in den Merkmalsraum abzubilden, verwenden wir hier eine explizite Merkmalsabbildung, die die Eingabedaten $x \in X$ in einen M-dimensionalen euklidischen Raum durch die Abbildung

$$\phi_M : X \to \mathbb{R}^M, \quad \phi_M(x) := \frac{1}{\sqrt{M}}\begin{pmatrix} \psi(x,\omega_1) \\ \vdots \\ \psi(x,\omega_M) \end{pmatrix}$$

projiziert, wobei ψ aus der Integraldarstellung (3.1) stammt. Dadurch lässt sich der Kern approximieren als

$$k(x,x') \approx k_M(x,x') = \phi_M(x)^\intercal \phi_M(x') = \langle \phi_M(x), \phi_M(x') \rangle.$$

Ein wesentlicher Vorteil der Methode besteht darin, dass die Dimension des Merkmalsraums begrenzt wird, wodurch sowohl der Speicherbedarf als auch die Rechenkomplexität im Vergleich zu klassischen Kernmethoden erheblich reduziert werden. Dies ermöglicht eine schnellere Anwendung linearer Verfahren, wie wir in Kapitel 7.1 genauer untersuchen werden.

5. Quasi-Monte-Carlo-Methode

5.1 Einführung in die Quasi-Monte-Carlo-Methode

Die Inhalte dieses Unterkapitels basieren auf [Nie92], Kapitel 1.3.

Um die Schwächen der MC-Methode zu umgehen, führen wir die Quasi-Monte-Carlo-Methode (QMC-Methode) ein. Die MC-Methode weist einen durchschnittlichen Fehler von $\frac{\sigma(f)}{\sqrt{M}}$ auf, was bedeutet, dass es M Punkte gibt, für die der Fehler kleiner oder gleich dieser Schranke ist. In der QMC-Methode versuchen wir, diese deterministischen Punktverteilungen zu finden.

Betrachten wir das Integral $\int_{[0,1]^d} f(x)\,\mathrm{d}x$ über dem d-dimensionalen Einheitswürfel, das wir mithilfe von (4.1) approximieren möchten. Anstelle zufälliger, gleichverteilter Punkte $x_1, \ldots, x_M \in [0,1]^d$ verwenden wir deterministische Punktmengen. Dies führt zu zwei zentralen Fragen: Wie lassen sich solche Punkte konstruieren? Und welchen Integrationsfehler erhalten wir?

Es zeigt sich, dass der Fehler mit $\mathcal{O}\left(\frac{\log(M)^d}{M}\right)$ eine deutlich bessere Schranke als bei der MC-Methode besitzt, insbesondere in niedrigen Dimensionen. Die QMC-Methode bietet zudem eine deterministische Fehlerschranke anstelle der probabilistischen Schranke der MC-Methode, was zu höherer Genauigkeit bei gleicher Anzahl von Funktionsauswertungen führt, ohne die Probleme der Zufallszahlengenerierung.

5.2 Folgen mit niedriger Diskrepanz

Die Inhalte dieses Unterkapitels basieren auf [Nie92], Kapitel 2.1.

Das Ziel ist es, den Integrationsfehler zu minimieren. Dazu müssen die deterministischen Punkte $x_1, \ldots, x_M$ möglichst gleichmäßig in $[0,1]^d$ verteilt sein. Doch wie kann man die Qualität der Verteilung dieser Punkte messen? Dazu führen wir die Stern-Diskrepanz ein.

Definition 5.1 (Stern-Diskrepanz)**.** Sei $\{x_i\}_{i=1}^M \subset [0,1]^d$ eine Punktmenge. Dann ist die Stern-Diskrepanz definiert als

$$\mathcal{D}^*(\{x_i\}_{i=1}^M) := \sup_{t \in [0,1]^d} \left| \mathrm{Vol}(J_t) - \frac{A(J_t; \{x_i\}_{i=1}^M)}{M} \right|$$

wobei $J_t := [0,t_1) \times [0,t_2) \times \cdots \times [0,t_d)$ ein Rechteck ist, $\mathrm{Vol}(J_t) := \prod_{i=1}^d t_i$ dessen Volumen angibt und

$$A(J_t; \{x_i\}_{i=1}^M) := |\{i \in \{1, \ldots, M\} : x_i \in J_t\}|$$

die Anzahl der Punkte x_i ist, die sich in dem Rechteck J_t befinden.

Die Stern-Diskrepanz vergleicht das tatsächliche Verhältnis der Punkte in einem Rechteck mit dem Volumen des Rechtecks. Je kleiner die Stern-Diskrepanz, desto gleichmäßiger sind die Punkte verteilt. Es gilt stets $\mathcal{D}^*(\{x_i\}_{i=1}^M) \in [0,1]$.

Wir sagen, dass eine Punktmenge $\{x_i\}_{i=1}^M$ niedrige Diskrepanz hat, falls $\mathcal{D}^*(\{x_i\}_{i=1}^M) = \mathcal{O}\left(\frac{(\log M)^d}{M}\right)$. Wir sagen, dass eine Folge $\{x_i\}_{i=1}^\infty$ niedrige Diskrepanz hat, falls $\mathcal{D}^*(\{x_i\}_{i=1}^M) = \mathcal{O}\left(\frac{(\log M)^d}{M}\right)$ für alle $M \geq 2$ gilt.

Im Folgenden werden verschiedene Folgen mit niedriger Diskrepanz vorgestellt. Dabei erläutern wir deren Konstruktion und bestimmen ihre Stern-Diskrepanz.

5.2.1 Die Van der Corput Folge

Die Inhalte dieses Unterkapitels basieren auf [Nie92], Kapitel 3.1.

Die Van der Corput Folge ist eine klassische Konstruktion einer eindimensionalen Folge mit niedriger Diskrepanz, die einfach zu generieren ist. Die Idee basiert auf der Darstellung von natürlichen Zahlen $n \geq 0$ in der b-adischen Zahlendarstellung, wobei $b \geq 2$ eine feste Basis ist.

Sei $b \geq 2$. Jede natürliche Zahl $n \geq 0$ kann in der Form

$$n = \sum_{i=0}^{\infty} a_i(n) b^i$$

geschrieben werden, wobei $a_i(n) \in \{0, 1, \dots b-1\}$ für alle $i \geq 0$. Da $a_i = 0$ für i genügend groß gilt, ist die Summe endlich. Die Van der Corput Folge wird durch die sogenannte radikal-inverse Funktion konstruiert.

Definition 5.2 (Radikal-Inverse Funktion)**.** Sei $b \geq 2$ eine ganze Zahl. Dann ist die radikal-inverse Funktion ϕ_b in der Basis b definiert als

$$\phi_b(n) = \sum_{i=0}^{\infty} a_i(n) b^{-i-1} \qquad \forall n \geq 2.$$

Durch die Anwendung der radikal-inversen Funktion ϕ_b auf alle natürlichen Zahlen entsteht die Van der Corput Folge.

Definition 5.3 (Van der Corput Folge)**.** Sei $b \geq 2$ eine ganze Zahl. Dann ist die Van der Corput Folge in der Basis b definiert als $\{x_n\}_{n=0}^{\infty}$, wobei $x_n = \phi_b(n)$ für alle $n \geq 0$.

Beispiel: Wir betrachten die Konstruktion der Van der Corput Folge in Basis $b = 2$ der Zahl $n = 14$. Dazu schreiben wir n in Binärdarstellung, „spiegeln“ diese und wandeln sie zurück in eine Dezimalzahl um, die anschließend durch b^k geteilt wird. Hierbei ist k die Anzahl der Stellen der Binärdarstellung von n.

1. Die Binärdarstellung von $n = 14$ ist:

   $$1110.$$

2. Diese Binärzahl wird „gespiegelt“, das heißt, die Ziffernreihenfolge wird umgekehrt:

   $$0111.$$

3. Die gespiegelte Binärzahl 0111 entspricht in der Dezimaldarstellung der Zahl 7.

4. Schließlich teilen wir 7 durch $2^4 = 16$:

   $$\phi_2(14) = \frac{7}{16}.$$

Damit ist die vierzehnte Zahl der Van der Corput Folge in Basis 2: $x_{14} = \frac{7}{16}$.

Für die radikal-inverse Funktion ϕ_b gilt stets $\phi_b(n) \in [0, 1)$ für alle $n \geq 0$. Die Stern-Diskrepanz der Van der Corput Folge ist von der Ordnung $\mathcal{O}\left(\frac{\log M}{M}\right)$.

5.2.2 Die Halton-Folge

Die Inhalte dieses Unterkapitels basieren auf [Nie92], Kapitel 3.1.

Die Halton-Folge (vgl. [Hal64]) ist eine Verallgemeinerung der eindimensionalen Van der Corput Folge auf mehrere Dimensionen. Die Konstruktion basiert auf der Wahl von d paarweise teilerfremden Zahlen als Basen, wobei d die Dimension der Punktmenge ist. Seien $b_1, \ldots, b_d$ paarweise teilerfremde ganze Zahlen. In der Regel werden hier die ersten d Primzahlen verwendet. Dann ist die d-dimensionale Halton-Folge $\{h_n\}_{n=0}^{\infty}$ in den Basen $b_1, \ldots, b_d$ definiert durch $h_n = (\phi_{b_1}(n), \ldots, \phi_{b_d}(n))^\intercal$ für alle $n \geq 0$, wobei ϕ_b die radikal-inverse Funktion in Basis b ist.

Die Halton-Folge bietet den Vorteil einer deterministischen Konstruktion mit niedriger Stern-Diskrepanz. Es gilt (vgl. [Ata04]):

$$\mathcal{D}^*(\{h_i\}_{i=1}^{M}) \leq C_H(d)\frac{(\log M)^d}{M}, \tag{5.1}$$

wobei $C_H(d)$ eine Konstante ist, die von der Dimension d abhängt.

Ein Nachteil der Halton-Folge liegt jedoch in der Wahl hoher Basen. In solchen Fällen kann die Gleichmäßigkeit der Punkteverteilung leiden. Die zweidimensionale Halton-Folge mit den Basen $(17, 19)$ weist für die ersten 16 Punkte eine perfekte lineare Korrelation auf, was in hohen Dimensionen zu einer schlechteren Gesamtverteilung führen kann (vgl. [KW97; GJ97]).

5.2.3 Die Sobol-Folge

Im Folgenden bezeichnen wir die Dimension mit s. Eine weitere Konstruktion von Punktmengen in $[0, 1]^s$ bietet die Sobol-Folge. Sie wurde ursprünglich von Sobol in [Sob67] für die Basis $b = 2$ eingeführt und später von Niederreiter in [Nie87] auf weitere Basen verallgemeinert. Niederreiter bezeichnet solche Folgen als (t, s)-Folgen in Basis b. Zur Definition dieser Folgen führen wir zunächst die Begriffe der elementaren Intervalle und (t, m, s)-Netze ein.

Die Inhalte dieses Unterkapitels basieren auf [Nie87] und [Nie92], Kapitel 4.

Definition 5.4 (Elementares Intervall)**.** Sei $b \geq 2$. Ein elementares Intervall in Basis b ist ein Intervall von der Form

$$E = \prod_{i=1}^{s} [a^{(i)} b^{-d_i}, (a^{(i)} + 1) b^{-d_i})$$

mit ganzen Zahlen $d_i \geq 0$ und $0 \leq a^{(i)} < b^{d_i}$ für $1 \leq i \leq s$.

Definition 5.5 ((t, m, s)-Netz)**.** Seien $b \geq 2$ und $0 \leq t \leq m$ ganze Zahlen. Ein (t, m, s)-Netz in Basis b ist eine Menge von b^m Punkten in $[0, 1]^s$, sodass $A(E; b^m) = b^t$ für jedes elementare Intervall E in Basis b mit $Vol(E) = b^{t-m}$. Hierbei ist $A(E; b^m)$ definiert wie in Definition 5.1.

Definition 5.6 ((t, s)-Folge)**.** Seien $b \geq 2$ und $t \geq 0$ ganze Zahlen. Eine Folge $x_1, x_2, \ldots$ von Punkten in $[0, 1]^s$ heißt (t, s)-Folge in Basis b, wenn für alle $k \geq 0$ und $m > t$ gilt, dass die Menge bestehend aus den Punkten x_n mit $kb^m \leq n < (k + 1)b^m$ ein (t, m, s)-Netz in Basis b ist.

Ein (t, m, s)-Netz kann so verstanden werden, dass die Punkte in $[0, 1]^s$ so gleichmäßig verteilt sind, dass in jedem elementaren Intervall E mit Volumen b^{t-m} exakt b^t Punkte liegen. Damit ist eine (t, s)-Folge so definiert, dass beliebige Teilmengen der Folge, die in aufeinanderfolgende Intervalle unterteilt sind, jeweils (t, m, s)-Netze darstellen. Je kleiner der Parameter t ist, desto gleichmäßiger ist die Verteilung der Punkte. Der minimale Wert für t hängt von der Dimension s ab.

Die Van der Corput Folge in Basis b ist ein Spezialfall und entspricht einer $(0, 1)$-Folge in Basis b. Die Stern-Diskrepanz einer (t, s)-Folge ist beschränkt durch:

$$\mathcal{D}^*(\{x_i\}_{i=1}^M) \leq C_S \cdot \frac{(\log M)^s}{M}$$

für $M \geq 2$.

Betrachten wir die Sobol-Folge in Basis $b = 2$. Im Folgenden wiederholen wir die Konstruktion der Sobol-Folge nach [DSHG08]. Dabei betrachten wir zunächst die Konstruktion einer eindimensionalen Sobol-Folge mit einer Präzision von N-Bits. Wir wählen ungerade Zahlen m_i, $i = 0, \ldots, N-1$, und definieren Richtungsvektoren durch

$$c_i = \frac{m_i}{2^i} = 0, c_{i1}c_{i2}c_{i3}\ldots,$$

wobei die Ziffern c_{ij} die Binärentwicklung von c_i bilden. Dann wird ein primitives Polynom

$$P(x) = x^d + a_1 x^{d-1} + \cdots + a_{d-1}x + 1$$

über $\mathbb{F}_2$ gewählt, mit Koeffizienten $a_i \in \{0, 1\}$. Dieses Polynom bestimmt die Richtungsvektoren c_i rekursiv durch

$$c_i = a_1 c_{i-1} \oplus a_2 c_{i-2} \oplus \cdots \oplus a_d c_{i-(d-1)} \oplus c_{i-d} \oplus [c_{i-d} \gg d],$$

wobei $\oplus$ die bitweise XOR-Operation bezeichnet und der letzte Term c_{i-d} um d Bits nach rechts verschoben wird, was einer Division durch 2^d entspricht. Sei n eine natürliche Zahl mit der binären Darstellung

$$n = b_k b_{k-1} \ldots b_2 b_1 \quad \text{mit } b_i \in \{0, 1\}.$$

Der n-te Punkt der Sobol-Folge wird berechnet durch

$$x_n = b_1 c_1 \oplus b_2 c_2 \oplus \cdots \oplus b_{k-1} c_{k-1} \oplus b_k c_k.$$

Diese Konstruktion gewährleistet eine gleichmäßige Verteilung der Punkte im Einheitswürfel und führt zu einer besseren Verteilung als Zufallszahlen. In höheren Dimensionen wird für jede Dimension ein eigenes primitives Polynom gewählt, wodurch eine multidimensionale Folge entsteht.

5.2.4 Vergleich der Halton-Folge mit der Sobol-Folge

Numerische Vergleiche aus [Moh19] zeigen, dass die Halton-Folge in niedrigen Dimensionen ähnliche Ergebnisse wie die Sobol-Folge liefert. Ab einer Dimension von $s > 1$ übertrifft die Sobol-Folge jedoch systematisch die Halton-Folge in Bezug auf die Genauigkeit der Integralapproximation. Zudem bleibt die Fehlerrate der Sobol-Folge stabil, während die Fehler bei der Halton-Folge mit wachsender Dimension tendenziell ansteigen. Ein weiterer Vorteil der Sobol-Folge ist ihre höhere Robustheit: Während die Punktverteilung der

Halton-Folge in hohen Dimensionen ungleichmäßiger wird, bleibt die Sobol-Folge auch in großen Dimensionen gut verteilt, was zu präziseren Ergebnissen führt.

Diese Beobachtungen bestätigen, dass die Wahl der QMC-Folge von der Problemstellung abhängt. Während die Halton-Folge in kleinen Dimensionen eine praktikable Alternative darstellt, ist die Sobol-Folge in höheren Dimensionen aufgrund ihrer besseren Verteilungseigenschaften vorzuziehen.

5.2.5 Weitere QMC-Folgen

In den numerischen Experimenten werden für die QMC-Methode neben der Halton-Folge und der Sobol-Folge auch die Digital-Net-Folge in Basis 2 und Gitterfolgen (engl. lattice sequences) (vgl. [DP10]) verwendet. Diese deterministischen Folgen bieten den Vorteil, dass sie eine nahezu gleichmäßige Abdeckung des Integrationsbereichs gewährleisten und dadurch zu einer schnelleren Konvergenz der Integralschätzungen führen, was ein wesentlicher Aspekt in der QMC-Methode ist.

Abbildung 5.1 zeigt beispielhaft 64 zufällig generierte Punkte im Vergleich zu den ersten 64 Punkten der jeweiligen QMC-Folge. Zur Erstellung der Abbildung wurden die Module `scipy.stats.qmc.Halton`, `scipy.stats.qmc.Sobol`, `qmcpy.DigitalNetB2` und `qmcpy.Lattice` verwendet.

Es wird deutlich, dass die zufälligen Punkte eine wesentlich ungleichmäßigere Verteilung aufweisen als die deterministischen Punkte der QMC-Folgen. Diese Beobachtung untermauert theoretische Resultate aus der Diskrepanztheorie, wonach geringere Diskrepanzen mit einer höheren Gleichmäßigkeit einhergehen.

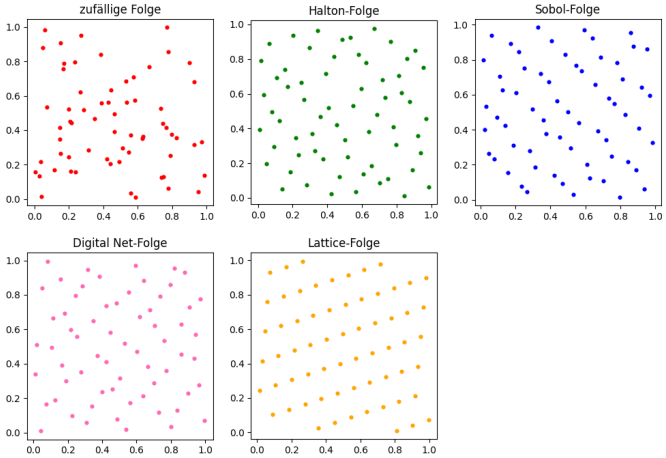

Abbildung 5.1: Beispiele für QMC-Folgen in $[0,1]^2$.

5.3 Integrationsfehler

Um den Fehler bei der Approximation eines Integrals zu analysieren, betrachten wir zwei wesentliche Faktoren: die Diskrepanz der verwendeten Punktmenge und die Eigenschaften des Integranden f. Zur Charakterisierung der Regularität von f führen wir die Variation im Sinne von Hardy und Krause ein.

Definition 5.7 (Variation im Sinne von Vitali (vgl. [Vit08], [Nie92]))**.** Sei f eine Funktion auf $[0,1]^d$ und $J \subset [0,1]^d$ ein Teilintervall. Sei $\Delta(f, J)$ eine alternierende Summe der Werte

von f in den Ecken von J, das heißt, die Funktionswerte von benachbarten Punkten haben entgegengesetzte Vorzeichen. Die Variation von f auf $[0,1]^d$ im Sinne von Vitali ist definiert als

$$V^{(d)}(f) = \sup_{\mathcal{P}} \sum_{J \in \mathcal{P}} |\Delta(f; J)|,$$

wobei das Supremum über alle Partitionen $\mathcal{P}$ von $[0,1]^d$ in Teilintervalle läuft.

Es gilt

$$V^{(d)}(f) = \int_0^1 \cdots \int_0^1 \left| \frac{\partial^d f}{\partial u_1 \dots \partial u_d} \right| \mathrm{d}u_1 \dots \mathrm{d}u_d,$$

falls die partiellen Ableitungen stetig auf $[0,1]^d$ sind (vgl. [Nie92]).

Definition 5.8 (Variation im Sinne von Hardy und Krause (vgl. [Nie92]))**.** Sei $1 \leq k \leq d$ und $1 \leq i_1 < i_2 < \cdots < i_k \leq d$. Sei $V^{(k)}(f; i_1, \dots, i_k)$ die Variation von f eingeschränkt auf

$$\{(u_1, \dots, u_d) \in [0,1]^d \mid u_j = 1 \text{ für } j \neq i_1, \dots, i_k\}$$

im Sinne von Vitali. Dann ist die Variation von f auf $[0,1]^d$ im Sinne von Hardy und Krause definiert als

$$V_{HK}(f) = \sum_{k=1}^{d} \sum_{1 \leq i_1 < i_2 < \cdots < i_k \leq d} V^{(k)}(f; i_1, \dots, i_k).$$

Die Variation im Sinne von Hardy und Krause ist eine Verallgemeinerung der totalen Variation (vgl. [Nie92]) und misst die Gleichmäßigkeit der Funktion hinsichtlich partieller Ableitungen in allen Koordinatenrichtungen.

Beispiel:

- Sei $f(x) = x^2$ auf dem Intervall $[0,1]$. Dann ist $V_{HK}(f) = 1$. Die Funktion hat keine starken Schwankungen und eine niedrige Variation.
- Sei $f(x) = \sin(100x)$ auf dem Intervall $[0,1]$. Dann ist

  $$V_{HK}(f) = 100 \int_0^1 |\cos(100x)| \, \mathrm{d}x = \infty.$$

  Die Funktion oszilliert schnell und hat unbeschränkte Variation.

Mithilfe der Variation im Sinne von Hardy und Krause können wir den Integrationsfehler der QMC-Methode abschätzen.

Satz 5.9 (Koksma-Hlawka Ungleichung (vgl. [Hla61]))**.** *Angenommen, $f : [0,1]^d \to \mathbb{R}$ hat beschränkte Variation im Sinne von Hardy und Krause $V_{HK}(f)$. Dann gilt für alle $x_1, \dots, x_M \in [0,1]^d$:*

$$\left| \int_{[0,1]^d} f(x) \, \mathrm{d}x - \frac{1}{M} \sum_{i=1}^{M} f(x_i) \right| \leq V_{HK}(f) \cdot \mathcal{D}^*(\{x_i\}_{i=1}^M)$$

Wie wir gesehen haben, gilt für die Halton-Folge $\mathcal{D}^*(\{x_i\}_{i=1}^M) = \mathcal{O}\left(\frac{(\log M)^d}{M}\right)$. Wenn wir die Halton-Folge $x_1, \dots, x_M \in [0,1]^d$ für die QMC-Methode verwenden und f eine Funktion mit beschränkter Variation im Sinne von Hardy und Krause ist, ergibt sich ein Integrationsfehler von der Ordnung $\mathcal{O}\left(\frac{(\log M)^d}{M}\right)$, was insbesondere in niedrigen Dimensionen besser ist als der Fehler der MC-Methode von $\mathcal{O}\left(\frac{1}{\sqrt{M}}\right)$.

6. Kernapproximation mit QMC-Features

Wenn wir die Approximation des Integrals aus (3.2) betrachten, erhalten wir

$$\left|k(x,x') - k_M(x,x')\right| \leq V_{HK}(f) \cdot \mathcal{D}^*(\{x_i\}_{i=1}^M).$$

Hierbei ist $x_1, \ldots, x_M \in [0,1]^{d+1}$, f der Integrand aus (3.2) und k_M ist definiert wie in (4.2). Wenn wir eine Folge mit niedriger Diskrepanz wählen, können wir den zweiten Faktor gut kontrollieren.

Allerdings weist der Integrand aus der Gleichung (3.2) eine unendliche Variation auf, wie in [Avr+16] gezeigt wurde. Insbesondere oszilliert der Integrand stark, wenn t sich den Rändern nähert, was dazu führt, dass der Integrand keine beschränkte Variation im Sinne von Hardy und Krause hat (vgl. [HSH24]).

Trotz dieser Herausforderung stellt sich die Frage, wie der Fehler der Integralapproximation dennoch sinnvoll abgeschätzt werden kann. Im Folgenden analysieren wir diese Problematik getrennt für translationsinvariante und nicht-translationsinvariante Kerne.

6.1 Translationsinvariante Kerne

Die Inhalte dieses Unterkapitels basieren auf [HSH24].

Wir formulieren zusätzliche Voraussetzungen, um die Anwendung von QMC-Methoden auf translationsinvariante Kerne zu ermöglichen.

QMC-Bedingung 1:

- Der Kern $k : X \times X \to \mathbb{R}$ ist translationsinvariant.
- Für die Verteilungsfunktion ϕ_i, $i = 1, \ldots, d$, definiert wie in Kapitel 3, gilt

  $$\frac{d}{dt}\phi_i^{-1}(t) \leq \frac{C_i}{\min(t, 1-t)}$$

  für eine Konstante $C_i > 0$ und $t \in (0,1)$.
- Die Menge X ist kompakt.

Die erste Bedingung stellt sicher, dass der Kern eine Integraldarstellung besitzt. Die zweite Bedingung kontrolliert die Ableitungen der Funktion ϕ_i^{-1}, wodurch die Variation des Integranden $f(t,b)$ insbesondere in der Nähe der Ränder von $[0,1]^d$ beschränkt wird. Die dritte Bedingung stellt sicher, dass X eine begrenzte Größe hat, wodurch die Fehlerabschätzung für die Kernapproximation (vgl. Satz 6.2) anwendbar wird.

Sei $u \subset \{1, \ldots, d+1\}$ eine nicht-leere Teilmenge. Wir definieren die partiellen Ableitungen des Integranden wie folgt:

$$\partial^u f(t,b) := \left(\prod_{j \in u} \frac{\partial}{\partial x_j}\right) f(x).$$

Lemma 6.1. *Sei $f : (0,1)^{d+1} \to \mathbb{R}$, sodass*

$$|\partial^u f(x)| \leq B \prod_{i \in u \setminus \{d+1\}} \frac{1}{\min(x_i, 1 - x_i)}$$

für ein $B > 0$ und alle nicht-leeren Teilmengen $u \subset \{1, \ldots, d+1\}$. Seien $h_1, \ldots, h_n$ die ersten n Punkte der $(d+1)$-dimensionalen Halton-Folge. Dann existiert eine Konstante $C(d)$, sodass für alle $n \geq 2$ gilt

$$\left| \int_{[0,1]^{d+1}} f(x) \, \mathrm{d}x - \frac{1}{n} \sum_{i=1}^{n} f(h_i) \right| \leq BC(d) \frac{(\log n)^{2d+1}}{n}.$$

Der Beweis ist in [HSH24] enthalten und beruht auf der Idee der sogenannten „Erweiterungen mit geringer Variation“ (vgl. [Owe06]). Dabei wird eine Funktion $\tilde{f}$ konstruiert, die eine beschränkte Variation aufweist und auf einer „großen“ Teilmenge $K_n \subset X$ mit der ursprünglichen Funktion f übereinstimmt. Ein zentraler Schritt des Beweises ist die Eigenschaft der Halton-Folge, den Rand des Einheitswürfels zu meiden (vgl. [Owe06]).

Satz 6.2. *Angenommen, k erfüllt QMC-Bedingung 1. Dann existiert ein $C > 0$, sodass für alle $x, x' \in X$ gilt*

$$|k_M(x, x') - k(x, x')| \leq \frac{C \log(M)^{2d+1}}{M}.$$

Beweis. Der Beweis basiert auf den Ideen aus [HSH24]. Hierbei wurden einige Details angepasst und erweitert.

Betrachten wir den Integranden f aus (3.2). Wir können f schreiben als

$$\begin{aligned} f(t, b) &= 2 \cos(x^\intercal \Phi^{-1}(t) + 2\pi b) \cos((x')^\intercal \Phi^{-1}(t) + 2\pi b) \\ &= \cos((x - x')^\intercal \Phi^{-1}(t)) + \cos((x + x')^\intercal \Phi^{-1}(t) + 4\pi b). \end{aligned}$$

Sei

$$D := \max_{x,y \in X, i \in \{1,\ldots,d\}} \{|x_i - y_i|, |x_i + y_i|\}.$$

Da die Menge X kompakt ist, ist D endlich. Für die partiellen Ableitungen von f gilt

$$\left| \frac{\partial}{\partial t_j} f(t, b) \right| \leq 2D \frac{d}{dt_j} \phi_j^{-1}(t_j)$$

und

$$\left| \frac{\partial}{\partial b} f(t, b) \right| \leq 4\pi.$$

Damit folgt

$$|\partial^u f(t, b)| \leq 4\pi (2D)^{|u \setminus \{d+1\}|} \prod_{i \in u \setminus \{d+1\}} \frac{d}{dt_i} \phi_i^{-1}(t_i)$$

für jedes nicht-leere $u \subset \{1, \ldots, d+1\}$. Mithilfe des Lemmas 6.1 folgt der Satz. □

Der Gauß-Kern und der Cauchy-Kern auf kompakten Mengen erfüllen QMC-Bedingung 1. Damit ergibt sich für diese Kerne bei der Approximation mit QMC-Methoden eine Fehlerordnung von $\mathcal{O}\left(\frac{(\log M)^{2d+1}}{M}\right)$, was insbesondere in niedrigen Dimensionen deutlich besser ist als die Fehlerordnung der MC-Methode von $\mathcal{O}\left(\frac{1}{\sqrt{M}}\right)$.

6.2 Nicht-translationsinvariante Kerne

Die Inhalte dieses Unterkapitels basieren auf [HSH24].

Wir wollen nun auch die Kerne betrachten, die nicht translationsinvariant sind und trotzdem eine Integraldarstellung besitzen. Dazu führen wir folgende Bedingungen ein.

QMC-Bedingung 2:

- Es existiert eine Funktion $\psi : X \times [0,1]^d \to \mathbb{R}$, sodass

$$k(x,x') = \int_{[0,1]^d} \psi(x,\omega)\psi(x',\omega)\,\mathrm{d}\omega$$

  für alle $x, x' \in X$.

- Die Funktion $g(\omega) = \psi(x,\omega)\psi(x',\omega)$ ist von beschränkter Variation im Sinne von Hardy und Krause $V_{HK}(g) \leq C_0$ für ein $C_0 > 0$.

Unter diesen Bedingungen folgt direkt aus der Koksma-Hlawka Ungleichung (Satz 5.9), dass für alle $x, x' \in X$ und $M \geq 2$ gilt

$$|k_M(x,x') - k(x,x')| \leq C_0 C_H(d) \frac{\log(M)^d}{M},$$

wenn wir die Halton-Folge wählen. Die Konstante $C_H(d)$ stammt aus der Stern-Diskrepanz der Halton-Folge (5.1). Bei der Wahl einer anderen Folge mit niedriger Diskrepanz ergibt sich eine entsprechend andere Konstante.

Beispiel: Min-Kern in einer Dimension. Sei $X = [0,1]$. Dann hat der Min-Kern die Darstellung

$$\begin{aligned} k(x,x') &= \min\{x,x'\} \\ &= \int_0^1 \mathbb{1}_{t<x}\mathbb{1}_{t<x'}\,\mathrm{d}t. \end{aligned} \tag{6.1}$$

Hier ist die Variation im Sinne von Hardy und Krause des Integranden $f(t) = \mathbb{1}_{t<x}\mathbb{1}_{t<x'} = \mathbb{1}_{t<\min\{x,x'\}}$ beschränkt durch 1. Damit erfüllt der Kern die QMC-Bedingung 2.

Beispiel: Produkt Kern. Angenommen

$$k_i(x,x') = \int_0^1 \psi_i(x,t)\psi_i(x',t)\,\mathrm{d}t$$

erfüllt für $i = 1,\ldots,d$ die QMC-Bedingung 2 mit $\psi_i(x,t) \leq \kappa_i$ für ein κ_i und alle x,t. Dann erfüllt

$$k(x,x') = \prod_{i=1}^d k_i(x_i,x_i') = \int_{[0,1]^d} \psi(x,t)\psi(x',t)\,\mathrm{d}t,$$

wobei $\psi(x,t) = \prod_{i=1}^d \psi_i(x_i,t_i)$, ebenfalls QMC-Bedingung 2. Hiermit können wir höherdimensionale Kerne konstruieren, für die QMC-Bedingung 2 ebenfalls gilt.

Damit ergibt sich für diese Kerne bei der Approximation mit QMC-Methoden eine Fehlerordnung von $\mathcal{O}\left(\frac{(\log M)^d}{M}\right)$.

7. Anwendung auf die Kern-Ridge-Regression

Wir haben die Kernfunktion mit der MC-Methode und der QMC-Methode approximiert, mit dem Ziel den hohen Rechenaufwand von $\mathcal{O}(N^3)$ und den Speicherbedarf von $\mathcal{O}(N^2)$ der Kern-Ridge-Regression (KRR) zu reduzieren. Im Folgenden wenden wir diese Approximationen auf die KRR an und analysieren deren Auswirkungen auf den Rechen- und Speicherbedarf.

7.1 Random Feature Kern-Ridge-Regression

Die Inhalte dieses Unterkapitels basieren auf [RR17].

Der Rechen- und Speicheraufwand bei der exakten KRR skaliert schlecht, insbesondere für große Datensätze. Um dieses Problem zu lösen, wird der Kern k durch Random Features approximiert. Wir setzen voraus, dass die Kernfunktion k durch ein Integral dargestellt werden kann, wie in (3.1):

$$k(x, x') = \int_\Omega \psi(x, \omega)\psi(x', \omega)\, \mathrm{d}\pi(\omega).$$

Diese Darstellung erlaubt es uns, den Kern durch zufällige, unabhängig und identisch verteilte Punkte $\{\omega_i\}_{i=1}^M$ aus π zu approximieren:

$$k_M(x, x') = \frac{1}{M}\sum_{i=1}^{M} \psi(x, \omega_i)\psi(x', \omega_i)$$

(vgl. (4.3)). Definieren wir die Abbildung $\phi_M : X \to \mathbb{R}^M$ durch

$$\phi_M(x) = \frac{1}{\sqrt{M}} \begin{pmatrix} \psi(x, \omega_1) \\ \vdots \\ \psi(x, \omega_M) \end{pmatrix} \in \mathbb{R}^M, \tag{7.1}$$

so lässt sich die Kernapproximation als Skalarprodukt in $\mathbb{R}^M$ schreiben:

$$k(x, x') \approx k_M(x, x') = \phi_M(x)^\intercal \phi_M(x') = \left\langle \phi_M(x), \phi_M(x') \right\rangle.$$

Durch Ersetzen des exakten Kerns k durch die Approximation k_M erhalten wir die approximierte Kernmatrix $K_M = [k_M(x_i, x_j)]_{i,j=1}^N$. Das resultierende Gleichungssystem lautet:

$$(K_M + N\lambda I_N)\alpha = y, \tag{7.2}$$

mit der regularisierten Lösung:

$$\hat{f}(x) = \sum_{i=1}^{N} \alpha_i k_M(x_i, x). \tag{7.3}$$

Die Lösung der RF-KRR kann durch die explizite Verwendung der Abbildung ϕ_M effizienter dargestellt werden, wie der folgende Satz zeigt.

Satz 7.1. *Seien $D = \{(x_i, y_i)\}_{i=1}^N$ gegebene Datenpunkte, $\psi(\cdot,\cdot)$ die durch die Integraldarstellung aus* (3.1) *definierte Funktion, und $\{\omega_i\}_{i=1}^M$ unabhängig und identisch verteilt. Sei ϕ_M definiert wie in* (7.1) *und definiere die Matrix*

$$\hat{\Phi}_M = \frac{1}{\sqrt{M}} \begin{pmatrix} \psi(x_1, \omega_1) & \dots & \psi(x_1, \omega_M) \\ \vdots & & \vdots \\ \psi(x_N, \omega_1) & \dots & \psi(x_N, \omega_M) \end{pmatrix} \in \mathbb{R}^{N \times M},$$

sowie den Vektor $y = (y_1, \dots, y_N)^\intercal \in \mathbb{R}^N$. Dann ist die approximative Lösung von (2.3) *mit der Kernapproximation k_M gegeben durch*

$$\hat{f}(x) = \phi_M(x)^\intercal (\hat{\Phi}_M^\intercal \hat{\Phi}_M + N\lambda I_M)^{-1} \hat{\Phi}_M^\intercal y. \tag{7.4}$$

Beweis. Wie wir gesehen haben, gilt

$$k_M(x, x') = \phi_M(x)^\intercal \phi_M(x') = \langle \phi_M(x), \phi_M(x') \rangle.$$

Damit können wir (7.3) schreiben als

$$\hat{f}(x) = \sum_{i=1}^N \alpha_i k_M(x_i, x) = \phi_M(x)^\intercal \hat{\Phi}_M^\intercal \alpha.$$

Ebenso können wir die approximierte Kernmatrix schreiben als

$$K_M = \hat{\Phi}_M \hat{\Phi}_M^\intercal.$$

Mit $\alpha = (K_M + N\lambda I_N)^{-1} y$ folgt

$$\hat{f}(x) = \phi_M(x)^\intercal \hat{\Phi}_M^\intercal (\hat{\Phi}_M \hat{\Phi}_M^\intercal + N\lambda I_N)^{-1} y.$$

Nun bleibt zu zeigen:

$$\hat{\Phi}_M^\intercal (\hat{\Phi}_M \hat{\Phi}_M^\intercal + N\lambda I_N)^{-1} = (\hat{\Phi}_M^\intercal \hat{\Phi}_M + N\lambda I_M)^{-1} \hat{\Phi}_M^\intercal.$$

Unter Verwendung der Identität (vgl. [Wel13], [Bis06] Anhang C):

$$PB^\intercal (BPB^\intercal + R)^{-1} = (P^{-1} + B^\intercal R^{-1} B)^{-1} B^\intercal R^{-1}$$

mit $P = I_M$, $B = \hat{\Phi}_M$ und $R = N\lambda I_N$ folgt

$$\begin{aligned} \hat{\Phi}_M^\intercal (\hat{\Phi}_M \hat{\Phi}_M^\intercal + N\lambda I_N)^{-1} &= \frac{1}{N\lambda} (I_M + \frac{1}{N\lambda} \hat{\Phi}_M^\intercal \hat{\Phi}_M) \hat{\Phi}_M^\intercal \\ &= (\hat{\Phi}_M^\intercal \hat{\Phi}_M + N\lambda I_M)^{-1} \hat{\Phi}_M^\intercal. \end{aligned}$$

Der Beweis der Identität folgt durch Multiplikation von rechts mit $(BPB^\intercal + R)$. □

Anhand von (7.4) wird ersichtlich, dass der Rechenaufwand der Random Feature Kern-Ridge-Regression $\mathcal{O}(NM^2 + M^3)$ und der Speicherbedarf $\mathcal{O}(NM)$ beträgt Dies ist eine deutliche Verbesserung gegenüber der exakten KRR, insbesondere wenn $M \ll N$.

7.2 QMC Feature Kern-Ridge-Regression

Die Inhalte dieses Unterkapitels basieren auf [HSH24].

Wie bereits gezeigt, führt die Verwendung von QMC-Features zu einem geringeren Approximationsfehler des Kerns im Vergleich zu Random Features (RF). Ein weiterer Vorteil der QMC-Methode besteht darin, dass sie reproduzierbare Ergebnisse liefert, da keine Zufallszahlen verwendet werden.

Betrachten wir Kernfunktionen in der Form von (3.2) und deren Approximation (4.2). Während der RF-Ansatz die Punkte $\{\omega_i\}_{i=1}^M$ gleichverteilt auf $[0,1]^d$ wählt, nutzt der QMC-Ansatz stattdessen eine Folge mit niedriger Diskrepanz auf $[0,1]^d$. Die Lösungsformel aus Satz (7.4) bleibt dabei unverändert gültig. Ebenso bleiben die Rechen- und Speicherkomplexitäten mit $\mathcal{O}(NM^2 + M^3)$ bzw. $\mathcal{O}(NM)$ erhalten.

Im Folgenden analysieren wir, in welchen Aspekten QMC-KRR der RF-KRR überlegen ist. Insbesondere untersuchen wir den Fehler bei der QMC-Approximation und bestimmen, wie die Anzahl der Punkte M gewählt werden muss, um mit der QMC-KRR denselben Fehler wie bei der RF-KRR zu erreichen.

7.3 Approximation des Integraloperators und der Kernmatrix

Die Inhalte dieses Unterkapitels basieren auf [HSH24; Avr+17].

In der KRR spielen der Integraloperator (vgl. Definition 2.4) und die Kernmatrix (vgl. Definition 2.2) eine zentrale Rolle. Diese Objekte hängen direkt von der Kernfunktion k ab, die oft durch eine approximierte Kernfunktion k_M ersetzt wird. Das Ziel ist es, den durch diese Approximation entstehenden Fehler zu kontrollieren und sicherzustellen, dass die Stabilität der Algorithmen erhalten bleibt. In diesem Abschnitt werden wir die Approximation des Integraloperators und der Kernmatrix definieren und den Fehler abschätzen.

Sei k eine Kernfunktion und k_M eine Approximation dieser Kernfunktion. Wie bereits gezeigt, gilt für die Approximation mit QMC-Features:

$$\sup_{x,x' \in X} |k(x,x') - k_M(x,x')| \leq C \frac{(\log M)^a}{M},$$

wobei C und a positive Konstanten sind.

Betrachten wir zunächst den Integraloperator. Die Approximation des Integraloperators kann definiert werden als:

$$L_M : L^2(P_{\boldsymbol{X}}) \to L^2(P_{\boldsymbol{X}})$$

$$\begin{aligned} L_M f(x) &:= \mathbb{E}_{\boldsymbol{X} \sim P_{\boldsymbol{X}}}[k(\boldsymbol{X}, x) f(\boldsymbol{X})] \\ &= \int_X k(x,z) f(z) \, \mathrm{d}P_{\boldsymbol{X}}(z). \end{aligned}$$

Unser Ziel ist es, die Norm der Differenz $\|L_M - L\|$ abzuschätzen, wobei $\|\cdot\|$ die Operatornorm ist. Dies ist möglich, wenn wir die Differenz $|k(x,x') - k_M(x,x')|$ kennen. Dazu formulieren wir die folgende Proposition.

Proposition 7.2 (vgl. [HSH24]). *Angenommen, die Kerne k und k_M erfüllen*

$$\sup_{x,x' \in X} |k(x,x') - k_M(x,x')| \leq C \frac{(\log M)^a}{M}$$

für $C, a > 0$. Dann gilt

$$\|L - L_M\| \leq C \frac{(\log M)^a}{M}.$$

Diese Proposition zeigt, dass der Fehler bei der Approximation des Integraloperators von derselben Größenordnung ist wie der Fehler bei der Approximation der Kernfunktion. Dies ist von zentraler Bedeutung für die Analyse der QMC-KRR, da der Integraloperator eine Schlüsselrolle in der Theorie der Kernmethoden spielt. Insbesondere ermöglicht diese Fehlerkontrolle die Herleitung theoretischer Garantien für die QMC-KRR, wie wir in Kapitel 7.4 genauer untersuchen werden.

In der KRR muss ein lineares Gleichungssystem mit der regularisierten Kernmatrix $K + \lambda I$ gelöst werden, wobei K die Kernmatrix des Kerns k ist. Durch die Approximation k_M von k können wir auch die Kernmatrix approximieren. Unser Ziel ist es, das Spektrum von $K + \lambda I_N$ und $K_M + \lambda I_N$ zu vergleichen. Dazu formulieren wir die folgende Proposition.

Proposition 7.3 (vgl. [HSH24]). *Angenommen, die Kerne k und k_M erfüllen*

$$\sup_{x,x' \in X} |k(x,x') - k_M(x,x')| \leq C \frac{(\log M)^a}{M}$$

für $C, a > 0$. Sei K die Kernmatrix von k und K_M die Kernmatrix von k_M bezüglich einer Menge $X_N = \{x_1, \ldots, x_N\}$. Seien $\lambda, \Delta > 0$. Falls $\frac{M}{(\log M)^a} \geq \frac{CN}{\Delta\lambda}$, dann gilt

$$(1 - \Delta)(K + \lambda I_N) \preceq K_M + \lambda I_N \preceq (1 + \Delta)(K + \lambda I_N). \tag{7.5}$$

Hierbei bedeutet $A \preceq B$ für Matrizen $A, B \in \mathbb{R}^{N \times N}$, dass $B - A$ positiv semidefinit ist.

Diese Proposition zeigt, dass $K_M + \lambda I_N$ eine Δ-Spektralapproximation von $K + \lambda I_N$ ist. Solche Spektralapproximationen sind entscheidend für statistische Garantien in der KRR. Die Proposition garantiert, dass die approximierte Kernmatrix K_M die Eigenschaften der exakten Kernmatrix K innerhalb eines kontrollierten Fehlerbereichs gut wiedergibt, solange genügend Features M verwendet werden. Dies ist besonders wichtig für numerisch stabile Lösungen in der KRR.

In [Avr+17] wurde gezeigt, dass für Random Features die Anzahl der benötigten Features M von der Ordnung $\frac{N}{\Delta^2\lambda}$ ist, um eine Δ-Spektralapproximation zu erreichen. Im Gegensatz dazu zeigt die obige Proposition, dass bei Verwendung von QMC-Features bereits M von der Ordnung $\frac{N}{\Delta\lambda}$ ausreicht. Die Gleichung (7.5) gilt, im Gegensatz zur Approximation mit Random Features, mit einer Wahrscheinlichkeit von 1. Dies deutet auf eine effizientere Approximation mit QMC-Features hin, da weniger Features benötigt werden, um eine vergleichbare Approximationsgüte zu erzielen.

7.4 Theoretische Ergebnisse für QMC-KRR

In diesem Abschnitt analysieren wir die theoretischen Eigenschaften der QMC-KRR basierend auf [HSH24]. Angenommen, $(\boldsymbol{X}, Y) \in X \times \mathbb{R}$ folgt der Verteilung $P_{\boldsymbol{X}Y}$ mit den Randverteilungen $P_{\boldsymbol{X}}$ und P_Y. Betrachten wir den erwarteten quadratischen Fehler

$$\mathcal{E}(f) = \mathbb{E}[(Y - f(\boldsymbol{X}))^2]$$

über der Funktionsklasse $\mathcal{H}$, wobei $\mathcal{H}$ der durch die Kernfunktion k definierte RKHS ist. Das Ziel ist es, $\mathcal{E}(f)$ unter Berücksichtigung der Regularisierung zu minimieren, um die Funktion

$$f_*(\boldsymbol{X}) = \mathbb{E}[Y|\boldsymbol{X}]$$

zu lernen. Um Aussagen über das überschüssige Risiko zu treffen, formulieren wir die folgenden Bedingungen:

KRR-Bedingung 1:

(i) Die Kernfunktion $k(x, x')$ ist stetig und hat eine Integraldarstellung gemäß (3.1) mit $|\psi(x, \omega)| \leq \kappa$ für ein $\kappa > 0$. Die Zufallsvariable $\boldsymbol{X}$ hat vollen Träger auf X und die Abbildung $\Omega \to L^2(P_{\boldsymbol{X}})$, $\omega \mapsto \psi(\cdot, \omega)$ ist stetig.

(ii) Das Maß π ist die Gleichverteilung auf $[0, 1]^p$ für ein $p \geq 1$ und die Integraldarstellung aus (3.1) wird mittels der QMC-Methode und (4.3) approximiert. Es gilt die Fehlerabschätzung:

$$\sup_{x,x' \in X} |k(x, x') - k_M(x, x')| \leq C \frac{(\log M)^a}{M}$$

für $C, a > 0$.

KRR-Bedingung 2: Die Verteilung von Y erfüllt die Bernstein-Bedingung: Es existieren Konstanten $\sigma, D > 0$, sodass für alle $k \geq 2$ gilt:

$$\mathbb{E}[|Y|^k|\boldsymbol{X}] \leq \frac{1}{2} k! \sigma^2 D^{k-2}.$$

KRR-Bedingung 3: Es existiert ein $r \in [\frac{1}{2}, 1]$, sodass $f_{\mathcal{H}} = L^r g$ für ein $g \in L^2(P_{\boldsymbol{X}})$, wobei $f_{\mathcal{H}}$ die Lösung von $\min_{f \in \mathcal{H}} \mathcal{E}(f)$ ist und L der Integraloperator aus Definition 2.4. Sei $R := \max\{\|g\|_{L^2(P_{\boldsymbol{X}})}, 1\}$.

Hierbei kann r als Maß für die Komplexität von $f_{\mathcal{H}}$ gesehen werden. Je größer r ist, desto glatter ist $f_{\mathcal{H}}$. Der Fall $r = \frac{1}{2}$ kann so verstanden werden, dass $f_{\mathcal{H}}$ in $\mathcal{H}$ existiert (vgl. [HSH24]).

Unter diesen Bedingungen erhalten wir das folgende Hauptresultat:

Satz 7.4 (vgl. [HSH24])**.** *Angenommen, KRR-Bedingungen 1, 2 und 3 gelten. Sei* $\lambda = \tilde{C} N^{-\frac{1}{2r+1}} \in (0, \frac{1}{e}]$ *und* $\hat{f}(x)$ *definiert wie in* (7.4)*. Dann reicht die Wahl von*

$$M = \frac{\log^a(1/\lambda)}{\lambda} = N^{\frac{1}{2r+1}} \log^a(N^{\frac{1}{2r+1}}/\tilde{C})/\tilde{C}$$

aus, um zu garantieren, dass für jedes $\delta \in (0, 1]$ *ein* N_0 *von Ordnung* $(\log \frac{1}{\delta})^{1+\frac{1}{2r}}$ *existiert, sodass für* $N \geq N_0$ *mit einer Wahrscheinlichkeit von mindestens* $1 - \delta$ *gilt:*

$$\mathcal{E}(\hat{f}) - \inf_{f \in \mathcal{H}} \mathcal{E}(f) \leq C_1 N^{-\frac{2r}{2r+1}} \log^2 \frac{6}{\delta}.$$

Hierbei ist C_1 *eine Konstante, die nur von* $\kappa, \sigma, D, R, r, \tilde{C}, C$ *und* a *abhängt.*

Der Satz gibt eine obere Schranke für das überschüssige Risiko $\mathcal{E}(\hat{f}) - \inf_{f \in \mathcal{H}} \mathcal{E}(f)$ der QMCF-KRR an. Die Beweisidee basiert auf einer Zerlegung des überschüssigen Risikos in

den Bias durch die Ridge-Regression, den Fehler durch Approximation mit zufälligen Features und den empirischen Fehler durch begrenzte Stichprobe. Der Beweis nutzt bekannte Ergebnisse für KRR und RF-KRR, wobei hier die spezifische Struktur von QMC-Features genutzt wird.

Bei Verwendung von Random Features zur Kernapproximation beträgt die Anzahl der benötigten Features

$$M \asymp N^{\frac{2r}{2r+1}} \log \frac{108\kappa^2 N}{\delta},$$

um ein überschüssiges Risiko von $\tilde{C}_1 N^{-\frac{2r}{2r+1}} \log^2 \frac{18}{\delta}$ zu erreichen, wohingegen bei der QMC-KRR nur

$$M = N^{\frac{1}{2r+1}} \log^a (N^{\frac{1}{2r+1}}/\tilde{C})/\tilde{C}$$

benötigt wird. Insbesondere für $r > \frac{1}{2}$ ergibt sich eine signifikante Verbesserung, da weniger Features M benötigt werden, um das gleiche überschüssige Risiko wie bei der RF-KRR zu erreichen.

8. Numerische Experimente

8.1 Approximation der Kernfunktion

Wie wir gesehen haben, erfüllt der Min-Kern die QMC-Bedingung 2 und der Gauß-Kern erfüllt QMC-Bedingung 1. Daher werden wir im Folgenden die Approximation dieser Kerne genauer betrachten. Wir visualisieren den Min-Kern und den Gauß-Kern sowie deren Approximation mithilfe der QMC- und MC-Methode. Analog zu [HSH24] approximieren wir den eindimensionalen Min-Kern $k(x, y) = \min\{x, y\}$ und den eindimensionalen Gauß-Kern $k(x, y) = \exp(-|x - y|^2)$ für $x, y \in [0, 1]$.

Zunächst berechnen wir die exakten Werte der Kernfunktion $k(x_i, y_j)$ für $i, j = 0, \ldots, 1000$, wobei $x_i = \frac{i}{1000}$ und $y_j = \frac{j}{1000}$. Anschließend berechnen wir die MC- und QMC-Approximationen $k_M(x_i, y_j)$ für $M = 25$. Dabei nutzen wir für den Min-Kern die Integraldarstellung aus (6.1) und für den Gauß-Kern die Integraldarstellung aus (3.2).

Für den Min-Kern ergibt sich:

$$k_M(x_i, y_j) = \frac{1}{M} \sum_{k=1}^{M} \mathbb{1}_{\omega_k < x_i} \mathbb{1}_{\omega_k < y_j},$$

wobei die $\{\omega_k\}_{k=1}^M$ entweder gleichverteilt auf $[0, 1]$ sind oder den ersten M Punkten der Halton-Folge in einer Dimension entsprechen. Für den Gauß-Kern berechnen wir:

$$k_M(x_i, y_j) = \frac{1}{M} \sum_{k=1}^{M} 2 \cos(x_i^\intercal \Phi^{-1}(t_k) + 2\pi b_k) \cos(y_j^\intercal \Phi^{-1}(t_k) + 2\pi b_k),$$

wobei $\{\omega_k\}_{k=1}^M$ mit $\omega_k = (t_k, b_k) \in \mathbb{R}^2$ entweder gleichverteilt auf $[0, 1]^2$ oder die ersten M Punkte der Halton-Folge in zwei Dimensionen sind. Hierbei bezeichnet $\Phi^{-1}(t)$ die Inverse der Verteilungsfunktion der Verteilung $\mathcal{N}(0, 2^2)$.

Die Halton-Folge ist deterministisch, sodass die Punkte $\omega_1, \ldots, \omega_M$ für jede Approximation $k(x_i, y_j)$ gleich bleiben. Im Rahmen der MC-Methode unterscheiden wir zwei Ansätze:

1. Feste Zufallspunkte: Eine einmalige Realisierung von $\omega_1, \ldots, \omega_M$ wird für alle Approximationen verwendet (vgl. [HSH24]).
2. Neu generierte Zufallspunkte: Für jedes Punktpaar (x_i, y_j) werden neue Zufallspunkte $\omega_1, \ldots, \omega_M$ gewählt.

In Abbildung 8.1 sind die exakten Kernfunktionen und die Approximationen dargestellt. Gelbe Punkte repräsentieren hohe Werte, während lila Punkte niedrige Werte zeigen.

Die QMC-Methode liefert immer identische und reproduzierbare Ergebnisse, da die verwendeten Punkte deterministisch sind. Im Gegensatz dazu erzeugt die MC-Methode bei jedem Durchlauf unterschiedliche Ergebnisse, da die Punkte zufällig gewählt werden, was zu einer Varianz in der Approximation führt. Der erste Ansatz mit festen zufälligen Punkten zeigt eine starke Abhängigkeit der Qualität der Approximation von der spezifischen Punktverteilung. Der zweite Ansatz führt zu einer zusätzlichen Variabilität, da für jedes Punktpaar neue zufällige Punkte generiert werden.

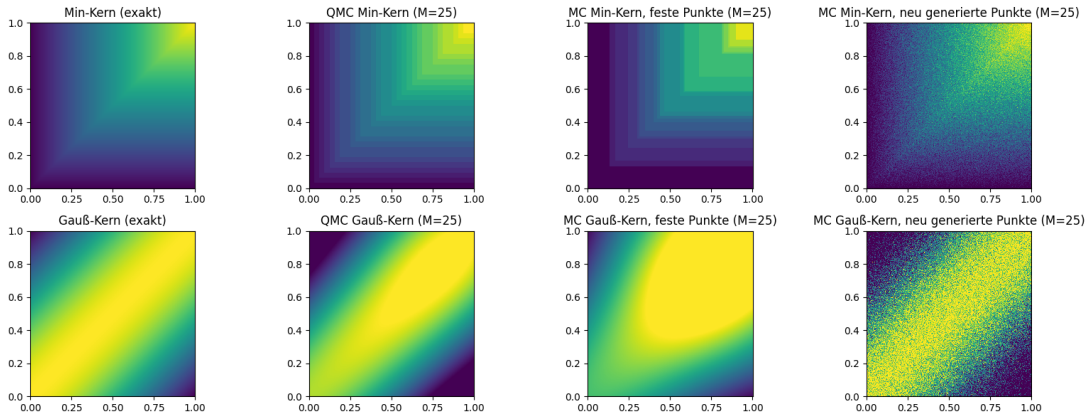

Abbildung 8.1: Kernapproximation mittels QMC- und MC-Methode mit festen sowie pro Approximation neu generierten Zufallspunkten.

Bereits bei $M = 25$ führt die QMC-Methode zu einer präziseren Approximation des Kerns als die beiden Varianten der MC-Methode. Um die Qualität der Approximation zu bewerten, betrachten wir den absoluten Fehler. Abbildung 8.2 zeigt die Fehlerdarstellungen.

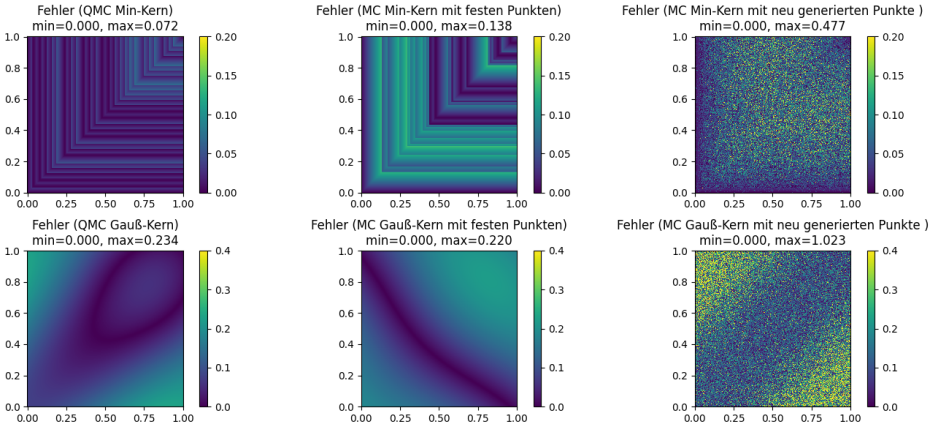

Abbildung 8.2: Absoluter Fehler der Kernapproximation mittels QMC- und MC-Methode mit festen sowie pro Approximation neu generierten Zufallspunkten.

Die Ergebnisse zeigen, dass die QMC-Approximation einen deutlich geringeren maximalen Fehler aufweist: $0,072$ für den Min-Kern und $0,234$ für den Gauß-Kern. Im Vergleich dazu betragen die Fehler bei der MC-Methode mit festen zufälligen Punkten $0,138$ bzw. $0,220$ und bei neu generierten Zufallspunkten $0,477$ bzw. $1,023$. Besonders auffällig ist der hohe Fehler bei der Methode mit neu generierten Zufallspunkten, was darauf zurückzuführen ist, dass zufällig gewählte Punkte stark variieren können, wodurch sowohl günstige als auch ungünstige Punktverteilungen entstehen.

Die Fehlerverteilung für den Min-Kern zeigt, dass der Fehler bei der QMC-Methode mit zunehmenden Werten von x und y ansteigt. Bei der MC-Methode mit festen Punkten ist der Fehler bei kleinen Werten von x und y tendenziell höher. Bei der Methode mit neu generierten Zufallspunkten ist der Fehler entlang der Koordinatenachsen sowie in der Umgebung des Punktes $x = y = 1$ gering, während er in den übrigen Bereichen tendenziell höher ausfällt.

Für den Gauß-Kern zeigt sich, dass der Fehler bei der QMC-Methode und der MC-Methode mit neu generierten zufälligen Punkten gering ist, wenn $|x-y|$ klein ist und ansteigt, wenn $|x-y|$ groß ist. Bei der QMC-Methode sowie der MC-Methode mit festen Punkten zeigt sich eine Struktur: Der Fehler ist in einem ringförmigen Bereich bzw. entlang einer bestimmten Linie besonders niedrig.

Diese Ergebnisse verdeutlichen, dass die Wahl der Methode einen erheblichen Einfluss auf die Approximationseigenschaften hat. Während die MC-Methode mit neu generierten Zufallspunkten eine größere Schwankung im Fehler aufweist, hängt die MC-Methode mit festen Zufallspunkten stark von den gewählten Punkten ab. Insgesamt zeigt sich, dass die QMC-Methode zu einer stabileren und genaueren Approximation führt.

8.2 Exakte KRR in einer Dimension

In diesem Abschnitt implementieren wir die exakte Kern-Ridge-Regression in einer Dimension und visualisieren diese, um das Verständnis der Methode zu verbessern. Wir betrachten den eindimensionalen Gauß-Kern, der wie folgt definiert ist:

$$k(x, x') = \exp\left(-\frac{|x - x'|^2}{2\sigma^2}\right).$$

Als Regressionsfunktion wählen wir $f(x) = 5x^2$. Die Trainingsdaten $\{(x_i, y_i)\}_{i=1}^{1000}$ werden zufällig erzeugt, wobei $x_i \sim \text{Unif}[0, 1]$ und sich die Zielwerte y_i aus $f(x_i)$ und einem Rauschterm $\epsilon \sim \mathcal{N}(0, (\frac{1}{2})^2)$ zusammensetzen, sodass $y_i = f(x_i) + \epsilon$.

Die Bandbreite σ des Gauß-Kerns wird anhand der Trainingsdaten berechnet, um die Skalierung des Kerns an die Daten anzupassen. Konkret verwenden wir dazu den Median der paarweisen Distanzen zwischen allen Trainingspunkten. Der Regularisierungsparameter wird auf $\lambda = 0,1$ gesetzt.

In Abbildung 8.3 ist das Ergebnis der KRR für dieses Szenario dargestellt. Die mit Rauschen behafteten Trainingsdaten sind als grüne Punkte visualisiert, die wahre Regressionsfunktion $f(x) = 5x^2$ ist als blaue Linie eingezeichnet und die von der KRR vorhergesagte Funktion ist durch die rote Linie repräsentiert.

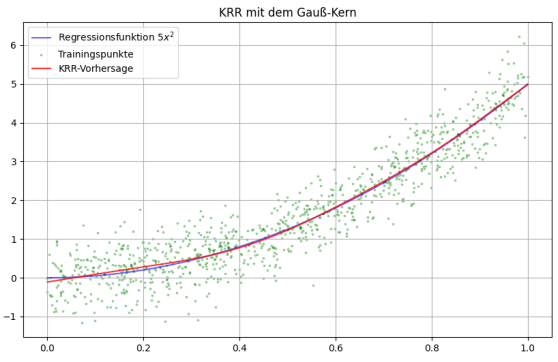

Abbildung 8.3: Exakte KRR mit dem Gauß-Kern.

Wie in Abbildung 8.3 zu sehen ist, approximiert die KRR die wahre Regressionsfunktion $f(x)$ sehr gut. Trotz des Rauschens in den Daten bleibt die Vorhersage der KRR stabil und liefert eine glatte Schätzung, die die zugrunde liegende Struktur der Funktion gut erfasst.

8.3 RF- und QMC-KRR im Vergleich zur exakten KRR

In diesem Abschnitt vergleichen wir die Approximationsgüte von RF-KRR, QMC-KRR und exakter KRR in höheren Dimensionen. Unser Ziel ist es, den Fehler zwischen den Vorhersagen der verschiedenen Methoden und synthetischen Testdaten zu analysieren. Dieser Vergleich orientiert sich an der Vorgehensweise in [HSH24].

Im Folgenden betrachten wir den Fall $r = 1$. Wir generieren synthetische Test- und Trainingsdaten anhand von $\boldsymbol{X} \sim \text{Unif}[0,1]^d$, $\epsilon \sim \mathcal{N}(0,1)$ und $Y = f(\boldsymbol{X}) + \epsilon$, wobei f die zugrundeliegende Regressionsfunktion ist.

Wir betrachten den Min-Kern

$$k(x, x') = \prod_{i=1}^{d} \min\{x_i, x_i'\}$$

und den Gauß-Kern

$$k(x, x') = \exp\left(-\frac{\|x - x'\|_2^2}{2\sigma^2}\right).$$

Dabei wird die Bandbreite σ des Gauß-Kerns anhand der Trainingsdaten $\boldsymbol{X} \sim \text{Unif}[0,1]^d$ als der Median der paarweisen Distanzen zwischen allen Trainingspunkten berechnet.

Für die Integraldarstellung des Min-Kerns nutzen wir in (4.3)

$$\psi(x, \omega) = \prod_{i=1}^{d} \mathbb{1}_{\omega_i < x_i},$$

wobei $\omega \in \mathbb{R}^d$ und für den Gauß-Kern

$$\psi(x, \omega) = \sqrt{2}\cos(x^\intercal \Phi^{-1}(t) + 2\pi b),$$

wobei $\omega = (t, b) \in \mathbb{R}^d \times \mathbb{R}$ und Φ^{-1} die Inverse der Verteilungsfunktion von $\mathcal{N}(0, \frac{1}{\sigma^2})$ ist.

Nach Satz 7.4 liegt die Regressionsfunktion in ran L^r. Für $r = 1$ hat sie die Form

$$\tilde{f}(x) = \int k(x, z) g(z) \, \mathrm{d}P_{\boldsymbol{X}}(z)$$

für ein $g \in L^2(P_{\boldsymbol{X}})$. Für den Min-Kern setzen wir

$$g(z) = \left(\prod_{i=1}^{d} z_i\right)^{\frac{1}{d}}.$$

Wir erhalten

$$\tilde{f}(x) = \left(\frac{d}{d+1}\right)^d \prod_{i=1}^{d} \left(x_i - \frac{d}{2d+1} x_i^{2+\frac{1}{d}}\right).$$

Für den Gauß-Kern setzen wir

$$g(z) = \exp\left(\frac{1}{2\sigma^2}\|z\|_2^2\right)$$

und erhalten

$$\tilde{f}(x) = \sigma^{2d} \exp\left(-\frac{1}{2\sigma^2}\|x\|_2^2\right) \prod_{i=1}^{d} \frac{\exp(x_i/\sigma^2) - 1}{x_i}.$$

Um das Signal-Rausch-Verhältnis zu kontrollieren, normalisieren wir $\tilde{f}$, indem wir $f(x) = C_{\tilde{f}} \cdot \tilde{f}(x)$ setzen, sodass $\mathbb{E}(f) = 5$ gilt. Den Regularisierungsparameter wählen wir gemäß Satz 7.4 als $\lambda = 0,25 \cdot N^{-\frac{1}{2r+1}}$, wobei N die Anzahl der Trainingspunkte bezeichnet. Dabei ist zu beachten, dass λ bereits von N abhängt, jedoch mit wachsendem N abnimmt. Um die Regularisierung konsistent an die Datenmenge anzupassen, multiplizieren wir in (7.4) λ mit N.

Wir betrachten die Dimensionen $d = 1, \ldots, 4$. Für jede Kombination aus Kern und Dimension generieren wir zunächst $2 \cdot 10^4$ Testdatenpunkte. Wir betrachten 500 Realisierungen von Trainingsdaten der Größe 10^4. Aus Laufzeitgründen wählen wir kleinere Testdaten und weniger Realisierungen als in [HSH24]. Im Fall des Gauß-Kerns berechnen wir die Bandbreite σ anhand des ersten Trainingsdatensatzes und berechnen für jede Dimension ein neues σ. Für jede der 500 Realisierungen trainieren wir das Modell, indem wir den Parameter

$$w = (\hat{\Phi}_M^\intercal \hat{\Phi}_M + n\lambda I_M)^{-1} \hat{\Phi}_M^\intercal y$$

gemäß (7.4) lernen, wobei $\hat{\Phi}_M$ von den Trainingsdaten abhängt. Die Inverse berechnen wir mithilfe von `numpy.linalg.solve`.

Anschließend sagen wir die Werte für den Testdatensatz vorher, indem wir für jedes x aus den Testdaten $f(x) = \phi_M(x)^\intercal w$ berechnen. Danach bestimmen wir die mittlere quadratische Abweichung (MQA) zwischen den vorhergesagten Werten und den Testwerten.

Wir vergleichen dabei die MC-Methode mit der QMC-Methode. Für die QMC-Methode verwenden wir die folgenden QMC-Folgen: die Halton-Folge, die Sobol-Folge, die Digital-Net-Folge und die Lattice-Folge. Zusätzlich zu der RF-KRR und der QMC-KRR führen wir die exakte KRR durch. Hierbei trainieren wir das Modell auf den Trainingsdaten und berechnen Vorhersagen für die Testdaten sowie die MQA. Die exakte KRR implementieren wir mithilfe von `sklearn.kernel_ridge.KernelRidge`.

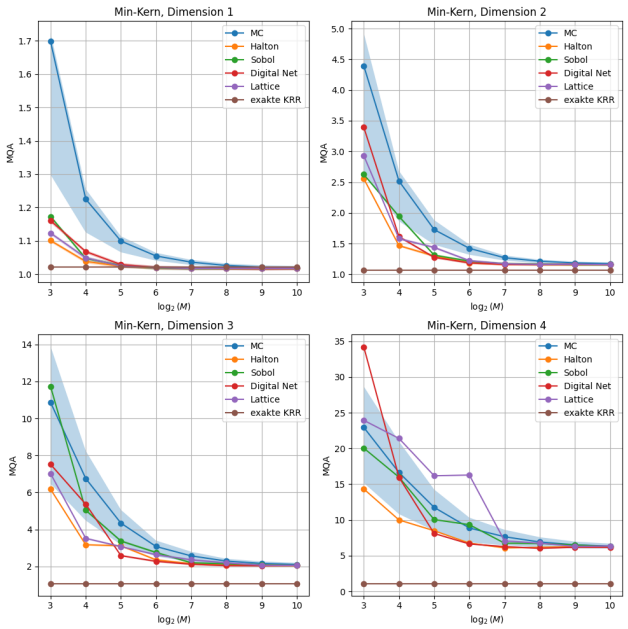

Abbildung 8.4: Mittlere quadratische Abweichungen der Methoden für den Min-Kern mit Mittelwert und den 25 %- und 75 %-Quantilen.

Die Abbildungen 8.4 und 8.5 visualisieren die MQA der Methoden für den Min-Kern und den Gauß-Kern. Die durchgezogene Linie gibt den Mittelwert über die 500 Realisierungen an, während die schattierten Bereiche die 25 %- und 75 %-Quantile darstellen.

Die Abbildung für den Min-Kern (Abbildung 8.4) zeigt, dass die MQA für die QMC-KRR im Allgemeinen niedriger ist als für die RF-KRR, welche auf der MC-Methode basiert. Besonders bei kleinen Werten von M ist der Vorteil der QMC-KRR gegenüber der RF-KRR deutlich ausgeprägt.

Ein auffälliger Unterschied zwischen den Methoden liegt in der Breite der Konfidenzbänder. Während die RF-KRR vergleichsweise große Konfidenzbänder aufweist, sind diese bei der QMC-KRR wesentlich kleiner. Dies ist darauf zurückzuführen, dass bei der RF-KRR nicht nur die Trainings- und Testdaten zufällig gewählt werden, sondern auch die Features für die Approximation zufällig generiert sind. Im Gegensatz dazu bleiben die Features bei der QMC-KRR deterministisch, sodass die einzige Quelle der Zufälligkeit in den Trainings- und Testdaten liegt (vgl. [HSH24]).

In einer Dimension überschneiden sich die Konfidenzbänder der QMC-KRR kaum mit denen der RF-KRR. In höheren Dimensionen ($d \geq 2$) werden die Unterschiede zwischen den Methoden jedoch geringer und die Konfidenzbänder der QMC-KRR und RF-KRR schneiden sich in einigen Fällen.

Für $d \in \{1, 2\}$ zeigt sich, dass die durchschnittliche MQA aller QMC-Folgen unterhalb der durchschnittlichen MQA der MC-Methode liegt. Dies bestätigt den Vorteil der QMC-Methode gegenüber der MC-Methode in niedrigen Dimensionen. Für höhere Dimensionen ($d \in \{3, 4\}$) nimmt dieser Vorteil jedoch ab. Insbesondere für $d = 4$ und kleinen Werten von M weist die Lattice-Folge eine auffällig hohe MQA auf. Die Digital-Net-Folge zeigt erst ab $M = 32$ stabile Ergebnisse, während die Halton-Folge über alle Dimensionen hinweg eine konsistent niedrigere MQA als die MC-Methode aufweist.

Ein allgemeiner Trend ist der Anstieg der MQA mit zunehmender Dimension. Dieser Effekt ist theoretisch erwartbar, da der Fehler der Ordnung $\mathcal{O}\left(\frac{(\log M)^d}{M}\right)$ ist und somit mit steigender Dimension d langsamer konvergiert. Gleichzeitig nimmt der Vorteil von QMC gegenüber RF mit steigender Dimension ab. In niedrigen Dimensionen sind die Unterschiede zwischen den Methoden klarer ausgeprägt, während in höheren Dimensionen einige QMC-Folgen keine signifikante Verbesserung gegenüber MC zeigen oder sogar schlechtere Ergebnisse liefern.

Die Abbildung für den Gauß-Kern (Abbildung 8.5) zeigt ähnliche Ergebnisse wie die Abbildung für den Min-Kern. Die QMC-KRR weist eine niedrigere MQA auf als die RF-KRR. Ein detaillierter Vergleich der Methoden zeigt, dass für $d \in \{2, 3\}$ die Sobol-Folge bei kleinen Werten von M schlechter abschneidet als die MC-Methode. Für $d = 4$ ist dies bei der Digital-Net-Folge der Fall. Für $d = 4$ liefert die Sobol-Folge die besten Ergebnisse.

Ein Unterschied zu den Ergebnissen des Min-Kerns besteht im Einfluss der Dimension. Während die MQA beim Min-Kern mit steigender Dimension stark ansteigt, fällt dieser Effekt beim Gauß-Kern deutlich geringer aus.

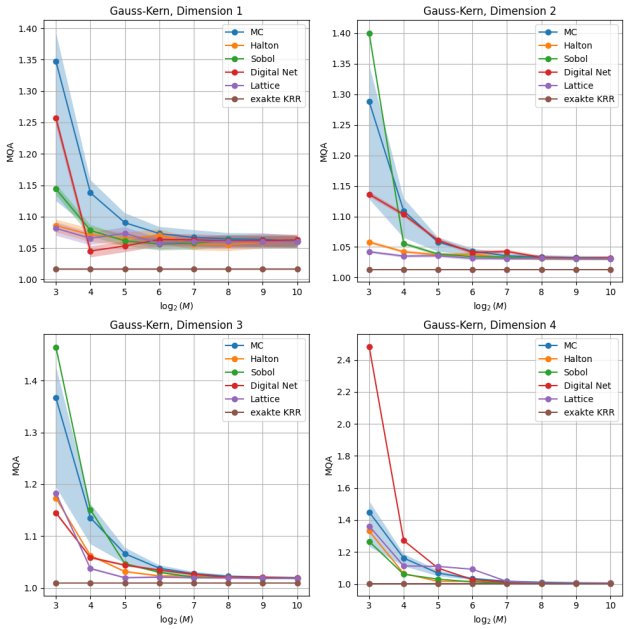

Abbildung 8.5: Mittlere quadratische Abweichungen der Methoden für den Gauß-Kern mit Mittelwert und den 25 %- und 75 %-Quantilen.

Untersuchung der Laufzeit

Wir analysieren die Laufzeiten der verschiedenen Methoden, indem wir die Zeit für das Training und die Vorhersage messen. Für jede der 500 Realisierungen erfassen wir die Laufzeit und berechnen den Mittelwert. Dabei betrachten wir die RF-KRR, die QMC-KRR mit der Halton-Folge sowie die exakte KRR. Die Ergebnisse sind in den Abbildungen A.1 und A.2 dargestellt und zeigen die Laufzeit abhängig von der Anzahl der Features M.

Die Ergebnisse verdeutlichen, dass die Approximationen mit der MC-Methode und der QMC-Methode für kleine Werte von M erheblich schneller sind als die exakte KRR. Mit zunehmendem M steigt die Laufzeit zwar an, bleibt jedoch selbst bei $M = 1024$ deutlich unter der der exakten KRR. Für den Min-Kern steigen die Laufzeiten mit der Dimension leicht an, wohingegen sie für den Gauß-Kern konstant bleiben. Zudem zeigt sich für den Min-Kern, dass die QMC-KRR schneller als die RF-KRR ist, während bei dem Gauß-Kern die beiden Approximationsansätze die gleiche Laufzeit haben. Nutzen wir Trainingsdaten der Größe $N = 1000$, sehen wir für $M = 1024$ keinen Vorteil in der Laufzeit. Diese Beobachtungen unterstreichen den erheblichen Rechenvorteil der RF-KRR und QMC-KRR gegenüber der exakten KRR, insbesondere für den Fall $M \ll N$.

Zusammenfassung der Ergebnisse

Insgesamt zeigt sich, dass QMC-KRR im Vergleich zu RF-KRR eine geringere MQA aufweist. Besonders in niedrigen Dimensionen sind die Vorteile von QMC-KRR deutlicher ausgeprägt. Zudem weisen die Approximationen eine deutlich geringere Laufzeit auf. Bei der QMC-Methode zeigen die Halton-, Sobol-, Digital-Net- und Lattice-Folgen unterschiedliche Eigenschaften. Die Halton-Folge liefert durchweg stabile Ergebnisse mit niedriger MQA. Für den Gauß-Kern in $d = 4$ weist die Sobol-Folge die niedrigste MQA auf.

Trotz kleinerer Testdaten und weniger Realisierungen stimmen unsere Ergebnisse mit denen aus [HSH24] überein.

8.4 Skalierung der Bandbreite

Der Gauß-Kern ist definiert als

$$k(x, x') = \exp\left(-\frac{\|x - x'\|^2}{2\sigma^2}\right),$$

wobei σ die Bandbreite des Kerns darstellt. Die Wahl von σ hat einen entscheidenden Einfluss auf die Ergebnisse der KRR.

Ein größerer Wert von σ führt zu einer breiteren Gauß-Funktion, sodass weiter entfernte Punkte stärker miteinander in Beziehung gesetzt werden. Ein kleinerer Wert von σ hingegen bewirkt, dass der Kern nur nahegelegene Punkte stark verbindet. Ist σ zu klein, kann dies zu Überanpassung führen, da das Modell stark an die Trainingsdaten angepasst wird. Umgekehrt kann ein zu großes σ zu einem zu steifen Modell mit geringer Anpassungsfähigkeit führen, was Unteranpassung verursacht. Die Wahl einer geeigneten Bandbreite σ ist daher entscheidend für die Balance zwischen Anpassung an die Daten und der Glattheit der Lösung.

In unserem Fall hängt ebenso die Regressionsfunktion von der Bandbreite σ ab. Ein größerer Wert von σ resultiert in einer glatteren Regressionsfunktion, während ein kleinerer Wert von σ zu einer stärkeren Variation führt.

Wie oben beschrieben berechnen wir die Bandbreite σ des Gauß-Kerns als Median der paarweisen Distanzen. Um den Einfluss von σ auf die MQA zu untersuchen, skalieren wir σ um den Faktor 2 bzw. $\frac{1}{2}$. Die Ergebnisse sind in den Abbildungen A.3 und A.4 dargestellt.

Die Abbildungen zeigen, dass eine größere Bandbreite σ mit einer niedrigeren MQA einhergeht. Obwohl in diesem Fall prinzipiell Unteranpassung möglich wäre, führt die starke Glättung der Regressionsfunktion zu einer insgesamt kleineren MQA.

Im Gegensatz dazu führt eine kleinere Bandbreite σ zu einer höheren MQA. Dies kann mit Überanpassung zusammenhängen: Das Modell passt sich stark an die Trainingsdaten an, verallgemeinert jedoch schlechter auf Testdaten, was zu einer erhöhten MQA führt.

Vergleicht man die Methoden der QMC-KRR und der RF-KRR, zeigt sich ein ähnliches Verhalten wie in Kapitel 8.3. Allerdings fällt der Vorteil der QMC-KRR im Vergleich zur RF-KRR bei der Skalierung mit beiden Faktoren hier geringer aus.

8.5 Vergleich der Methoden in höheren Dimensionen

Aufgrund der Erkenntnisse in Kapitel 8.3 konzentrieren wir uns im Folgenden auf höhere Dimensionen und betrachten die exakte KRR sowie die RF-KRR und QMC-KRR mit der Halton- und Sobol-Folge, ähnlich zu den Untersuchungen in [HSH24]. Der Ablauf entspricht dem in Kapitel 8.3, jedoch erweitern wir die Betrachtung auf die Dimensionen $d \in \{5, 10, 15, 20\}$ für den Min-Kern und $d \in \{5, 10, 25, 50\}$ für den Gauß-Kern.

Zunächst betrachten wir den Min-Kern. Im vorherigen Kapitel haben wir festgestellt, dass die MQA in der Dimension $d = 4$ deutlich höher ausfällt als in einer Dimension. Nun untersuchen wir das Verhalten der MQA für $d \in \{5, 10, 15, 20\}$. Die Ergebnisse in Abhängigkeit von der Anzahl der Features M sind in Abbildung 8.6 dargestellt.

Die Ergebnisse zeigen, dass für $d = 5$ die MQA insgesamt höher ist als für $d = 4$. Während die Halton-Folge hier bessere Ergebnisse als die RF-KRR liefert, schneidet die Sobol-Folge

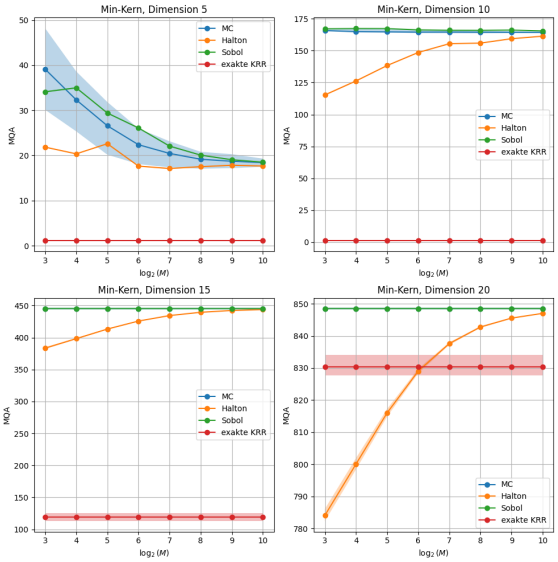

Abbildung 8.6: Mittlere quadratische Abweichungen für den Min-Kern in den Dimensionen $d \in \{5, 10, 15, 20\}$.

schlechter ab. Zudem sinkt die MQA mit zunehmendem M. In der Dimension $d = 10$ bleibt die MQA für die exakte KRR vergleichsweise gering. Allerdings weisen die Approximationsmethoden mit MC, Halton und Sobol eine sehr hohe MQA auf. Für die Halton-Folge steigt die MQA sogar mit zunehmender Feature-Anzahl M. In dieser Dimension erreichen die MQA-Werte für die Approximationsmethoden bereits dreistellige Werte, was auf eine deutliche Verschlechterung hinweist. Für $d = 15$ beobachten wir ein ähnliches Verhalten. Allerdings liegt hier auch die MQA der exakten KRR im dreistelligen Bereich. In $d = 20$ verschlechtert sich die Situation weiter, sodass sämtliche MQA-Werte über 700 liegen.

Diese Ergebnisse zeigen, dass die QMC-KRR und RF-KRR mit dem Min-Kern ab einer Dimension von $d = 10$ keine brauchbaren Resultate mehr liefern. Ab $d = 15$ gilt dies sogar für die exakte KRR. Somit ist der Min-Kern für die KRR in hohen Dimensionen nicht geeignet.

Betrachten wir jetzt den Gauß-Kern in den Dimensionen $d \in \{5, 10, 25, 50\}$. In den bisher untersuchten Dimensionen $d \in \{1, 2, 3, 4\}$ blieb die MQA in etwa in derselben Größenordnung. Mit der Erweiterung der Analyse auf höhere Dimensionen wollen wir insbesondere die MQA der Halton- und der Sobol-Folge untersuchen. Dabei betrachten wir, wie sich die MQA in Abhängigkeit von der Anzahl der verwendeten Features verhält. Die Ergebnisse dieser Untersuchung sind in Abbildung 8.7 dargestellt.

In der Dimension $d = 5$ zeigt sich der deutliche Vorteil der QMC-Methode gegenüber der MC-Methode, wie auch in Kapitel 8.3. Insbesondere ist hier zu erkennen, dass die Sobol-Folge bessere Ergebnisse als die Halton-Folge liefert. Für $d = 10$ lassen sich dieselben Beobachtungen machen. In der Dimension $d = 25$ fällt auf, dass die Halton-Folge deutlich schlechter abschneidet als die MC-Methode, während die Sobol-Folge weiterhin überlegen bleibt. Dieses Verhalten setzt sich auch für $d = 50$ fort.

Damit bestätigen sich die Ergebnisse aus [HSH24] bezüglich der Halton-Folge. Zudem zeigt sich, dass in hohen Dimensionen die Sobol-Folge deutlich besser für die QMC-Approximation geeignet ist als die Halton-Folge, was die Resultate aus Kapitel 5.2.4 untermauert.

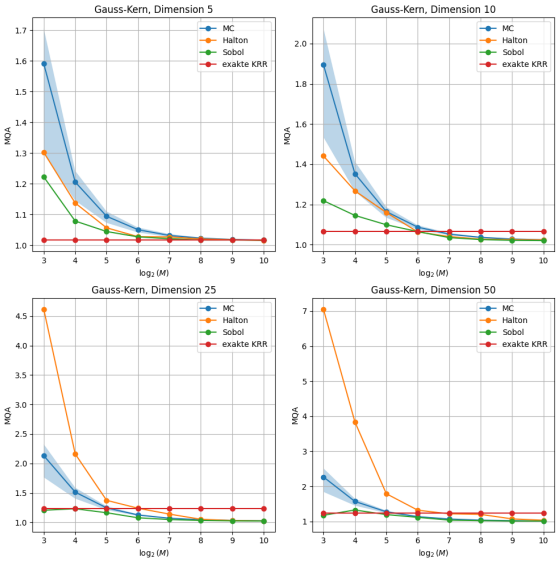

Abbildung 8.7: Mittlere quadratische Abweichungen für den Gauß-Kern in den Dimensionen $d \in \{5, 10, 25, 50\}$.

In der Praxis wird häufig zunächst eine Dimensionsreduktion durchgeführt (vgl. [HSH24]), um die wesentlichen Merkmale aus den Daten zu extrahieren. Anschließend erfolgt die Kernapproximation in dem dadurch entstehenden niedrigdimensionalen Raum. Diese Vorgehensweise ermöglicht es, bei realen Datensätzen mit einer Vielzahl an Features gezielt auf die relevanten Informationen zu fokussieren. In Kapitel 8.7 wird dieser Ansatz anhand eines realen Datensatzes weiter untersucht.

8.6 Vergleich der Methoden bei reduzierter Regularität

In Kapitel 8.3 wurde bereits der Fall $r = 1$ untersucht, während Satz 7.4 für $r \in [\frac{1}{2}, 1]$ gilt und insbesondere für $r > \frac{1}{2}$ eine Verbesserung liefert. In diesem Kapitel betrachten wir den Grenzfall $r = \frac{1}{2}$ analog zu [HSH24], der einem weniger glatten Szenario entspricht, da r als Glattheitsparameter interpretiert werden kann (vgl. [HSH24]).

Die Regressionsfunktion liegt in ran $L^{\frac{1}{2}}$, was, wie in [HSH24] gezeigt, gleichbedeutend mit dem RKHS $\mathcal{H}$ ist. Dementsprechend wählen wir für den Min-Kern die Regressionsfunktion

$$\tilde{f}(x) = 2k(1_d, x) - k\left(\frac{3}{4}1_d, x\right) + 2k\left(\frac{1}{2}1_d, x\right) - k\left(\frac{1}{4}1_d, x\right)$$

für den Gauß-Kern

$$\tilde{f}(x) = k\left(\frac{1}{3}1_d, x\right) + k\left(\frac{2}{3}1_d, x\right).$$

Hierbei bezeichnet 1_d den d-dimensionalen Vektor, dessen Einträge alle Eins sind. In beiden Fällen gilt somit $\tilde{f} \in$ ran $L^{\frac{1}{2}}$. Zusätzlich normalisieren wir, sodass $\mathbb{E}(f) = 5$. Den weiteren Ablauf führen wir analog zum Fall $r = 1$ durch.

Die Ergebnisse sind in den Abbildungen A.5 und A.6 dargestellt. Auch hier zeigt sich, dass die QMCF-KRR eine geringere MQA aufweist als die RF-KRR. Das beobachtete Verhalten entspricht dem für $r = 1$. Insgesamt sind die MQA für alle Methoden hier niedriger als in dem Fall $r = 1$. Zudem erweist sich die Halton-Folge in den Dimensionen $d \in \{1, 2, 3, 4\}$ als stabiler als die Sobol-Folge. Die Ergebnisse stimmen mit denen in [HSH24] überein.

8.7 Vergleich der Methoden auf einem realen Datensatz

Im Folgenden analysieren wir die exakte Kern-Ridge-Regression (KRR), die Random Feature KRR (RF-KRR) sowie die Quasi-Monte-Carlo KRR (QMC-KRR) anhand eines realen Datensatzes.

Der Wine-Datensatz (vgl. [Cor+09]) besteht aus Eigenschaften von Weinen sowie Bewertungen durch menschliche Experten. Er umfasst insgesamt 6 497 Proben (1 599 rote und 4 898 weiße Weine).

Jede Weinprobe ist durch 11 chemische Eigenschaften charakterisiert, darunter Zuckergehalt, Dichte, pH-Wert, Sulfatgehalt und Alkoholgehalt. Zusätzlich wurde jeder Wein von mindestens drei professionellen Testern auf einer Skala von 0 bis 10 bewertet, wobei die mittlere Bewertung als endgültige Qualitätsbewertung dient. Das Ziel ist es, mithilfe der KRR-Verfahren die Weinbewertung auf Basis der chemischen Eigenschaften vorherzusagen.

Wir nutzen k-fache Kreuzvalidierung mit $k = 6$ (vgl. [Bis06], Kapitel 1.3). Der Datensatz wird in 6 gleich große Teilmengen unterteilt, wobei in jeder Iteration eine Teilmenge als Testdaten dient, während die restlichen 5 Teilmengen als Trainingsdaten genutzt werden. Dadurch wird sichergestellt, dass jede Beobachtung sowohl als Trainings- als auch als Testdaten verwendet wird.

Für die KRR nutzen wir den Gauß-Kern, wie bereits in den vorherigen Experimenten. Der Parameter σ wird in jeder Iteration anhand der Trainingsdaten berechnet. Wir setzen den Regularisierungsparameter $\lambda = 0,0001$. Für die QMC-KRR nutzen wir die Halton- und die Sobol-Folge.

In Abbildung 8.8 sind die mittleren quadratischen Abweichungen (MQA) in Abhängigkeit von der Feature-Anzahl M dargestellt. Es werden die durchschnittliche MQA über die 6 Iterationen sowie die 25 %- und 75 %-Quantile gezeigt.

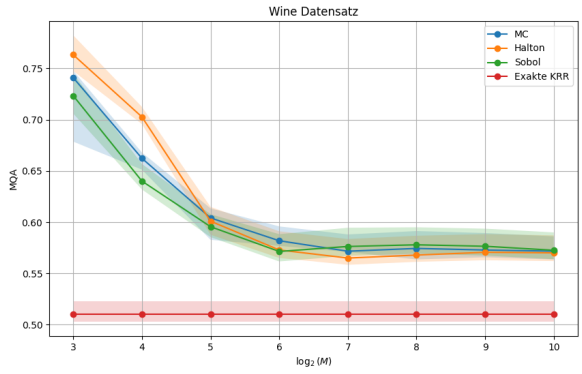

Abbildung 8.8: Mittlere quadratische Abweichungen der KRR mit Gauß-Kern auf realen Daten und 6-facher Kreuzvalidierung.

Aus Abbildung 8.8 lassen sich folgende Beobachtungen ableiten:

- Bis $M = 2^6$ ist die MQA für die Sobol-Folge geringer als für die Halton-Folge.
- Für $M = 2^3$ und $M = 2^4$ ist die MQA für die Halton-Folge höher als für die MC-Methode.
- Ab $M = 2^6$ sind alle drei Methoden ähnlich gut.

Insgesamt liefert die QMC-KRR mit der Sobol-Folge die besten Ergebnisse der drei Approximationsmethoden.

Da der Vorteil der QMC-Methode insbesondere in niedrigeren Dimensionen deutlich wird, führen wir nun eine Dimensionsreduktion mithilfe der Hauptkomponentenanalyse (PCA) durch. Zur weiteren Untersuchung reduzieren wir die Anzahl der Dimensionen von 11 auf 5 und führen anschließend die gleichen Experimente erneut durch. Die Ergebnisse sind in Abbildung 8.9 dargestellt.

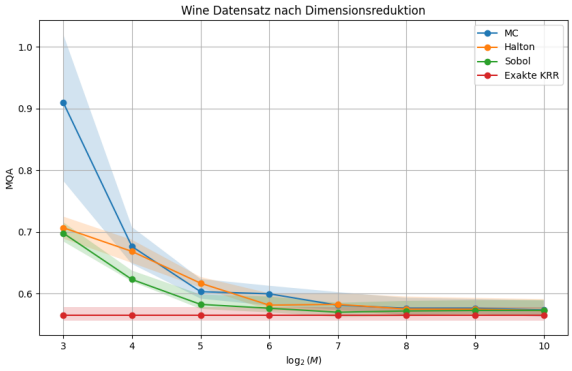

Abbildung 8.9: Mittlere quadratische Abweichungen der KRR mit Gauß-Kern auf realen Daten mit 6-facher Kreuzvalidierung und vorheriger Dimensionsreduktion auf 6 Dimensionen.

Aus Abbildung 8.9 ergeben sich folgende Erkenntnisse:

- Die Ergebnisse mit der QMC-Methode weisen eine geringere MQA als die MC-Methode auf und zeigen eine kleinere Konfidenzbänder.
- Die Sobol-Folge erzielt eine bessere Genauigkeit als die Halton-Folge.

Verglichen mit der vorherigen Analyse ohne Dimensionsreduktion zeigt sich für die RF-KRR und die exakte KRR, dass die MQA nach der Dimensionsreduktion höher ist. Das deutet darauf hin, dass durch die Dimensionsreduktion gewisse Informationen verloren gehen. Für die QMC-KRR sind die MQA nahezu unverändert. Ein weiterer interessanter Aspekt ist, dass nach der Dimensionsreduktion die Approximationen näher an das exakte Ergebnis heranrücken als ohne Dimensionsreduktion.

Insgesamt zeigt sich der Vorteil der QMC-KRR im Vergleich zur RF-KRR, besonders in niedrigen Dimensionen, was die Experimente auf den synthetischen Daten sowie die theoretischen Ergebnisse bestätigt.

9. Fazit

Ziel dieser Arbeit war es, die theoretischen Grundlagen und experimentellen Ergebnisse aus [HSH24] zur Kernapproximation mittels der QMC-Methode zu untersuchen.

Dazu haben wir zunächst die Theorie der Kernapproximation mittels Random Features und der QMC-Methode systematisch aufgearbeitet, wobei insbesondere die in [HSH24] dargestellten theoretischen Resultate im Mittelpunkt standen. Anschließend wurden die in [HSH24] durchgeführten numerischen Experimente reproduziert. Darüber hinaus haben wir die Implementierung erweitert. Dazu zählten Untersuchungen unter Einsatz verschiedener QMC-Folgen, detaillierte Laufzeitmessungen, die Analyse des Einflusses der Bandbreite des Gauß-Kerns σ sowie Tests an einem zusätzlichen realen Datensatz.

Die Ergebnisse unserer Experimente, die analog zu [HSH24] durchgeführt wurden, stimmen, trotz des Einsatzes kleinerer Testdatensätze und weniger Realisierungen, weitgehend mit den Ergebnissen aus [HSH24] überein. Insbesondere konnte der Vorteil der QMC-KRR gegenüber der RF-KRR sowohl bei synthetischen als auch bei realen Daten, vor allem in niedrigen Dimensionen, eindeutig nachgewiesen werden. Zudem bestätigt sich die erwartete geringere Varianz der QMC-Methode. Unsere weiterführenden Experimente zeigten jedoch auch, dass die erzielten Resultate stark von der Wahl der QMC-Folge abhängen. Während die Sobol-Folge auch in hohen Dimensionen stabile Ergebnisse liefert, ergab sich bei der in [HSH24] untersuchten Halton-Folge in höheren Dimensionen ein deutlicher Nachteil gegenüber der RF-KRR. Darüber hinaus beeinflusst die Wahl der Bandbreite σ signifikant die Resultate. Die Laufzeitmessungen bestätigen sowohl die theoretischen Vorhersagen als auch den Vorteil der QMC-Methode.

Zusammenfassend bestätigen unsere Untersuchungen die in [HSH24] dargestellten Vorteile der QMC-Methode, insbesondere hinsichtlich einer verbesserten Laufzeit und einer schnelleren Konvergenzordnung der QMC-KRR im Vergleich zur RF-KRR sowie einem ausgeprägten Vorteil in niedrigdimensionalen Szenarien. Die Ergebnisse unterstreichen dabei nicht nur die Effizienz der QMC-Methode, sondern weisen auch auf eine hohe Reproduzierbarkeit der Resultate hin.

Hinsichtlich weiterer Optimierungspotenziale und Erweiterungsmöglichkeiten bietet sich insbesondere die vertiefte Untersuchung der optimalen Wahl von QMC-Folgen in Abhängigkeit von der Dimension der Daten an. Auch der Einfluss einer vorherigen Dimensionsreduktion sollte weiter analysiert werden, um den Trade-off zwischen Rechenaufwand und Genauigkeit noch präziser bestimmen zu können. Ferner könnten zukünftige Arbeiten die Anwendung der QMC-Methode auf weitere Kernfunktionen und in unterschiedlichen praktischen Anwendungsgebieten untersuchen, um so die allgemeine Anwendbarkeit und Robustheit der Methode weiter zu validieren.

Diese Erweiterungen bieten vielversprechende Ansätze für zukünftige Forschungen und tragen dazu bei, die Einsatzmöglichkeiten und die theoretische Fundierung der Kernapproximation mittels der QMC-Methode weiter zu vertiefen.

A. Ergänzende Diagramme

Laufzeitmessung

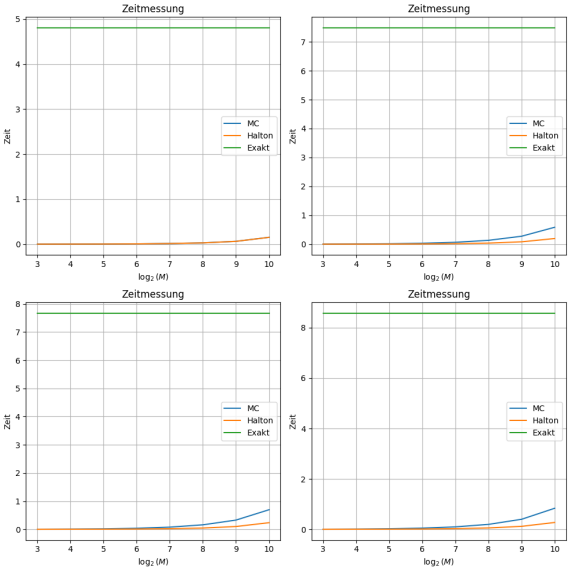

Abbildung A.1: Laufzeit der KRR für den Min-Kern in Abhängigkeit von M, verglichen für die exakte Methode, QMC und MC.

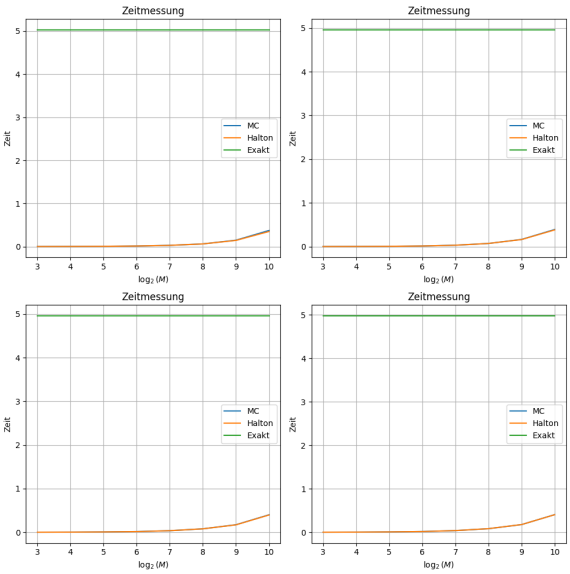

Abbildung A.2: Laufzeit der KRR für den Gauß-Kern in Abhängigkeit von M, verglichen für die exakte Methode, QMC und MC.

Skalierung des Parameters σ

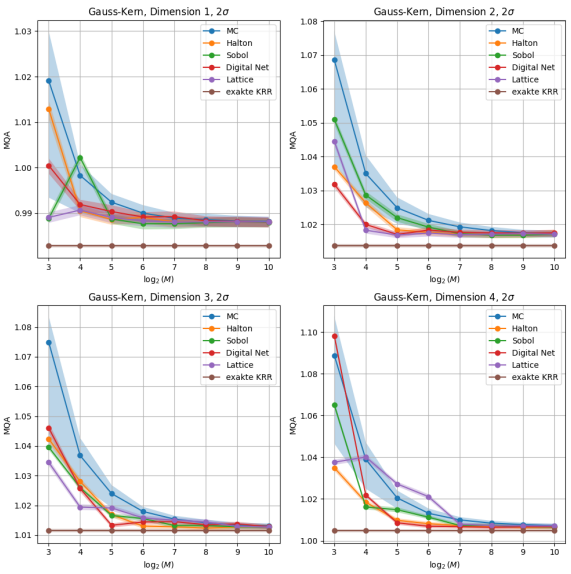

Abbildung A.3: Mittlere quadratische Abweichungen für die KRR mit dem Gauß-Kern und skaliertem Parameter 2σ.

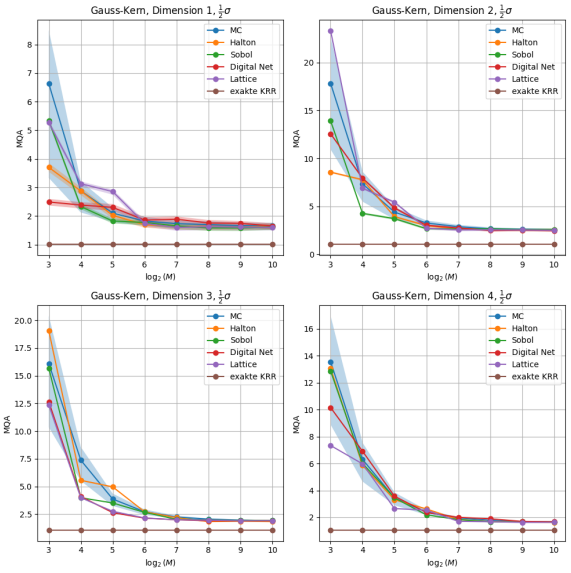

Abbildung A.4: Mittlere quadratische Abweichungen für die KRR mit dem Gauß-Kern und skaliertem Parameter $\frac{1}{2}\sigma$.

Grenzfall $r = \frac{1}{2}$ mit reduzierter Regularität

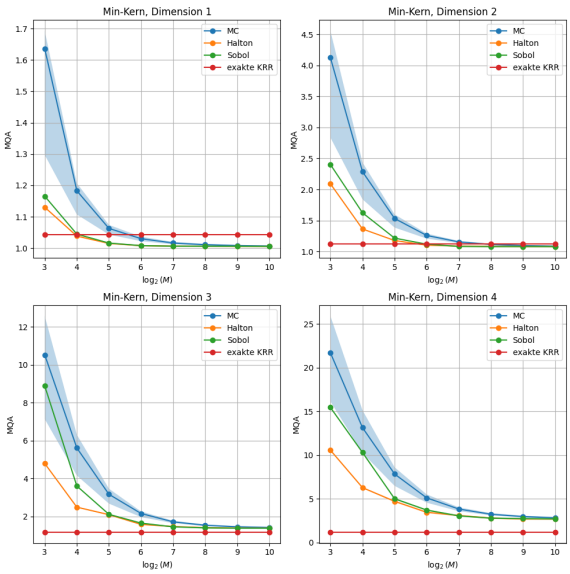

Abbildung A.5: Mittlere quadratische Abweichungen für den Min-Kern in dem Fall $r = \frac{1}{2}$.

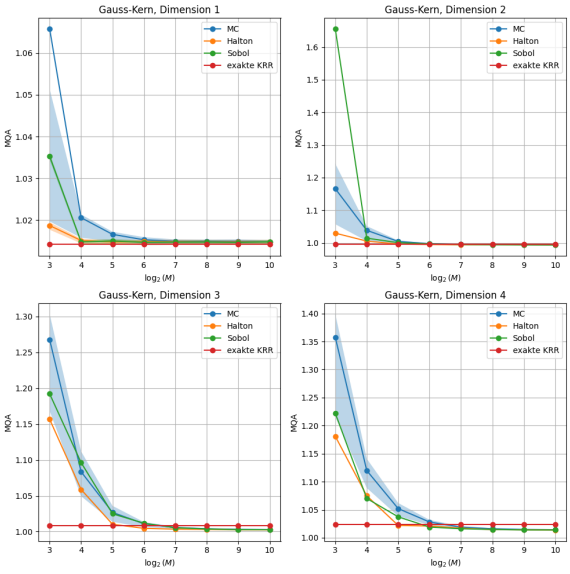

Abbildung A.6: Mittlere quadratische Abweichungen für den Gauß-Kern in dem Fall $r = \frac{1}{2}$.

Literatur

[Ata04] E. Atanassov. „On the discrepancy of the Halton sequences“. In: *Mathematica Balkanica. New Series* 18 (Jan. 2004).

[Avr+16] Haim Avron u. a. „Quasi-Monte Carlo Feature Maps for Shift-Invariant Kernels“. In: *Journal of Machine Learning Research* 17.120 (2016), S. 1–38.

[Avr+17] Haim Avron u. a. „Random Fourier features for kernel ridge regression: Approximation bounds and statistical guarantees“. In: *International conference on machine learning*. PMLR. 2017, S. 253–262.

[Bis06] Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Berlin, Heidelberg: Springer-Verlag, 2006.

[Boc33] S. Bochner. „Monotone Funktionen, Stieltjessche Integrale und harmonische Analyse“. In: *Mathematische Annalen* 108.1 (1. Dez. 1933), S. 378–410.

[Cor+09] Paulo Cortez u. a. „Modeling wine preferences by data mining from physicochemical properties“. In: *Decision Support Systems* 47.4 (2009). Smart Business Networks: Concepts and Empirical Evidence, S. 547–553.

[DP10] Josef Dick und Friedrich Pillichshammer. *Digital Nets and Sequences: Discrepancy Theory and Quasi–Monte Carlo Integration*. Cambridge University Press, 2010.

[DSHG08] Ishaan L. Dalal, Deian Stefan und Jared Harwayne-Gidansky. „Low discrepancy sequences for Monte Carlo simulations on reconfigurable platforms“. In: *2008 International Conference on Application-Specific Systems, Architectures and Processors*. 2008, S. 108–113.

[FHH17] Jeremy Ferwerda, Jens Hainmueller und Chad Hazlett. „Kernel-Based Regularized Least Squares in R (KRLS) and Stata (krls)“. In: *Journal of Statistical Software* 79 (Juli 2017).

[GJ97] Silvio Galanti und Alan Jung. „Low-discrepancy sequences: Monte Carlo simulation of option prices“. In: *The Journal of Derivatives* 5.1 (1997), S. 63–83.

[Hal64] J. H. Halton. „Algorithm 247: Radical-inverse quasi-random point sequence“. In: *Commun. ACM* 7.12 (Dez. 1964), S. 701–702.

[Hla61] Edmund Hlawka. „Funktionen von beschränkter Variatiou in der Theorie der Gleichverteilung“. In: *Annali di Matematica Pura ed Applicata* 54.1 (1. Dez. 1961), S. 325–333.

[HSH24] Zhen Huang, Jiajin Sun und Yian Huang. „Quasi-Monte Carlo Features for Kernel Approximation“. In: *Proceedings of the 41st International Conference on Machine Learning*. Hrsg. von Ruslan Salakhutdinov u. a. Bd. 235. Proceedings of Machine Learning Research. PMLR, Juli 2024, S. 20134–20165.

[KW97] Ladislav Kocis und William J. Whiten. „Computational investigations of low-discrepancy sequences“. In: *ACM Trans. Math. Softw.* 23.2 (Juni 1997), 266–294.

[Moh19] Nadia A Mohammed. „Comparing halton and sobol sequences in integral evaluation“. In: *Zanco Journal of Pure and Applied Sciences* 31.1 (2019).

[Nie87] Harald Niederreiter. „Point sets and sequences with small discrepancy“. In: *Monatshefte für Mathematik* 4 (1. Dez. 1987), S. 273–337.

[Nie92] Harald Niederreiter. *Random Number Generation and Quasi-Monte Carlo Methods*. Society for Industrial und Applied Mathematics, 1992.

[Owe06] Art B. Owen. „Halton Sequences Avoid the Origin“. In: *SIAM Review* 48.3 (2006), S. 487–503.

[Pre+25] Preethi u. a. „Optimizing Polynomial and Regularization Techniques for Enhanced Housing Price Prediction Accuracy“. In: *SN Computer Science* 6.2 (2025), S. 96.

[RR07] Ali Rahimi und Benjamin Recht. „Random Features for Large-Scale Kernel Machines“. In: *Advances in Neural Information Processing Systems*. Hrsg. von J. Platt u. a. Bd. 20. Curran Associates, Inc., 2007.

[RR17] Alessandro Rudi und Lorenzo Rosasco. „Generalization properties of learning with random features“. In: *Proceedings of the 31st International Conference on Neural Information Processing Systems*. 2017, 3218–3228.

[RW06] Carl Edward Rasmussen und Christopher K. I. Williams. *Gaussian processes for machine learning*. English. Cambridge, MA: MIT Press, 2006.

[SC08] Ingo Steinwart und Andreas Christmann. *Support vector machines*. eng. Information science and statistics. New York, NY: Springer, 2008.

[Seo+22] Hoon Seo u. a. „Scaling multi-instance support vector machine to breast cancer detection on the BreaKHis dataset“. In: *Bioinformatics* 38.S1 (2022), S. i92–i100.

[Sob67] I.M Sobol'. „On the distribution of points in a cube and the approximate evaluation of integrals“. In: *USSR Computational Mathematics and Mathematical Physics* 7.4 (1967), S. 86–112.

[SS02] Bernhard Scholkopf und Alexander J. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press, 2002.

[SSM98] Bernhard Schölkopf, Alexander Smola und Klaus-Robert Müller. „Nonlinear Component Analysis as a Kernel Eigenvalue Problem“. In: *Neural Computation* 10 (Juli 1998), S. 1299–1319.

[Vit08] G. Vitali. „Sui gruppi di punti e sulle funzioni di variabili reali.“ Italian. In: *Torino Atti* 43 (1908), S. 229–246.

[Wel13] Max Welling. „Kernel ridge regression“. In: *Max Welling's classnotes in machine learning* (2013), S. 1–3.