



TECHNISCHE UNIVERSITÄT
BERGAKADEMIE FREIBERG

Die Ressourcenuniversität. Seit 1765.

Grundlagen GIS

Sommersemester 2024

Prof. Christian Gerhards
Arbeitsgruppe Geomathematik und Geoinformatik

OPAL:

<https://bildungsportal.sachsen.de/opal/auth/RepositoryEntry/19758415873?6>

Institut für Geophysik und Geoinformatik - TU Bergakademie Freiberg

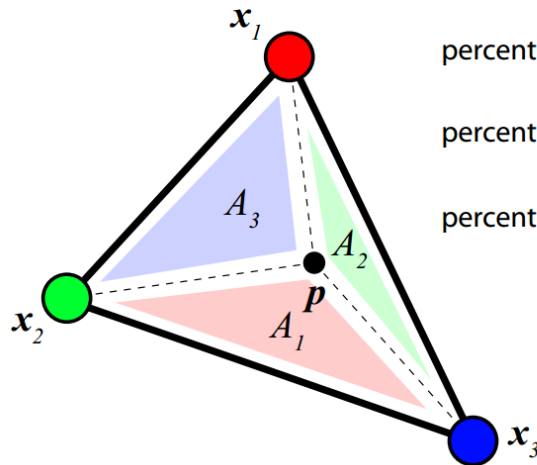
Lineare Interpolation auf Dreiecken

Gegeben seien die Datentupel (x_i, y_i) , (x_j, y_j) , (x_k, y_k) und sei x ein Punkt innerhalb eines Dreiecks $\mathcal{C}(x_i, x_j, x_k)$. Punkt x lässt sich als Linearkombination

$$x = \lambda_i x_i + \lambda_j x_j + \lambda_k x_k$$

ausdrücken, gleiches gilt für den unbekanntem Messwert am Punkt x :

$$f(x) = \lambda_i y_i + \lambda_j y_j + \lambda_k y_k$$



$$\left. \begin{aligned} \text{percent red} &= \frac{A_1}{A} = \lambda_1 \\ \text{percent green} &= \frac{A_2}{A} = \lambda_2 \\ \text{percent blue} &= \frac{A_3}{A} = \lambda_3 \end{aligned} \right\} \text{ "barycentric coordinates" }$$

Value at p :

$$(A_1 x_1 + A_2 x_2 + A_3 x_3) / A$$

$$\sum_i \lambda_i = 1$$

Lineare Interpolation auf Dreiecken

Gegeben seien die Datentupel (x_i, y_i) , (x_j, y_j) , (x_k, y_k) und sei x ein Punkt innerhalb eines Dreiecks $\mathcal{C}(x_i, x_j, x_k)$. Punkt x lässt sich als Linearkombination

$$x = \lambda_i x_i + \lambda_j x_j + \lambda_k x_k$$

ausdrücken, gleiches gilt für den unbekanntem Messwert am Punkt x :

$$f(x) = \lambda_i y_i + \lambda_j y_j + \lambda_k y_k$$

Die Koeffizienten / Gewichte werden als **baryzentrische Koordinaten** von x (in 2D). Seien die Koordinaten von $x = (x^{(1)}, x^{(2)})$, dann können die Koeffizienten wie folgt ausgedrückt werden

$$\lambda_i = \frac{\det(A_i)}{\det(A)}, \quad \lambda_j = \frac{\det(A_j)}{\det(A)}, \quad \lambda_k = \frac{\det(A_k)}{\det(A)}$$

mit den Matrizen

$$A = \begin{pmatrix} x_i^{(1)} & x_j^{(1)} & x_k^{(1)} \\ x_i^{(2)} & x_j^{(2)} & x_k^{(2)} \\ 1 & 1 & 1 \end{pmatrix}, A_i = \begin{pmatrix} x^{(1)} & x_j^{(1)} & x_k^{(1)} \\ x^{(2)} & x_j^{(2)} & x_k^{(2)} \\ 1 & 1 & 1 \end{pmatrix}, A_j = \begin{pmatrix} x_i^{(1)} & x^{(1)} & x_k^{(1)} \\ x_i^{(2)} & x^{(2)} & x_k^{(2)} \\ 1 & 1 & 1 \end{pmatrix}, A_k = \begin{pmatrix} x_i^{(1)} & x_j^{(1)} & x^{(1)} \\ x_i^{(2)} & x_j^{(2)} & x^{(2)} \\ 1 & 1 & 1 \end{pmatrix}$$

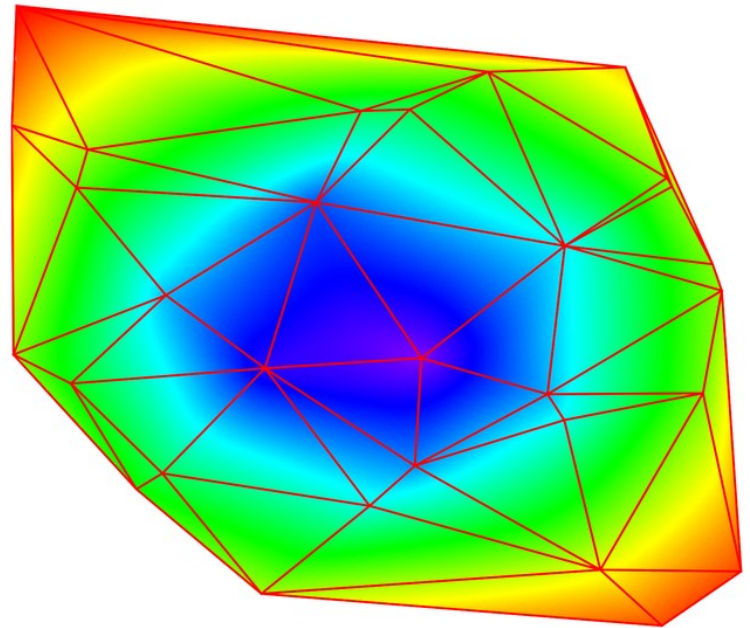
Lineare Interpolation auf Dreiecken

Gegeben seien die Datentupel (x_i, y_i) , (x_j, y_j) , (x_k, y_k) und sei x ein Punkt innerhalb eines Dreiecks $\mathcal{C}(x_i, x_j, x_k)$. Punkt x lässt sich als Linearkombination

$$x = \lambda_i x_i + \lambda_j x_j + \lambda_k x_k$$

ausdrücken, gleiches gilt für den unbekanntem Messwert am Punkt x :

$$f(x) = \lambda_i y_i + \lambda_j y_j + \lambda_k y_k$$



Lineare Interpolation auf Dreiecken

Gegeben seien die Datentupel (x_i, y_i) , (x_j, y_j) , (x_k, y_k) und sei x ein Punkt innerhalb eines Dreiecks $\mathcal{C}(x_i, x_j, x_k)$. Punkt x lässt sich als Linearkombination

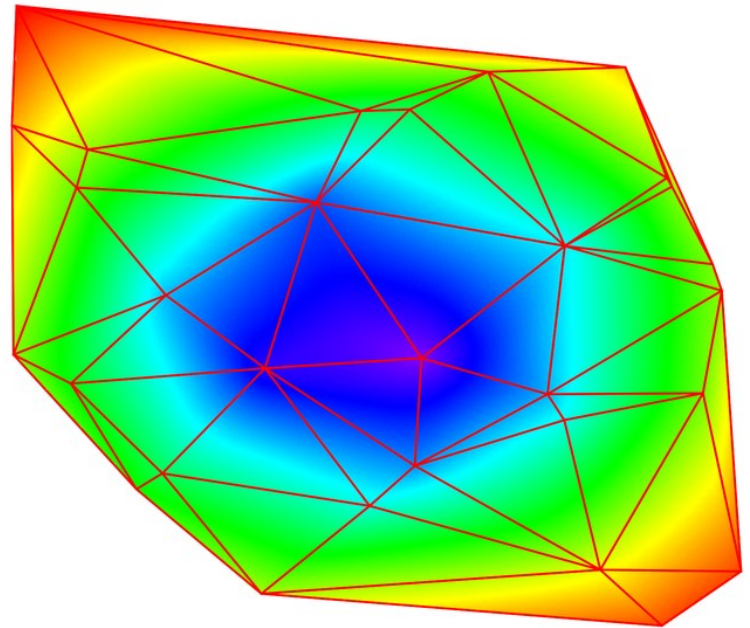
$$x = \lambda_i x_i + \lambda_j x_j + \lambda_k x_k$$

ausdrücken, gleiches gilt für den unbekanntem Messwert am Punkt x :

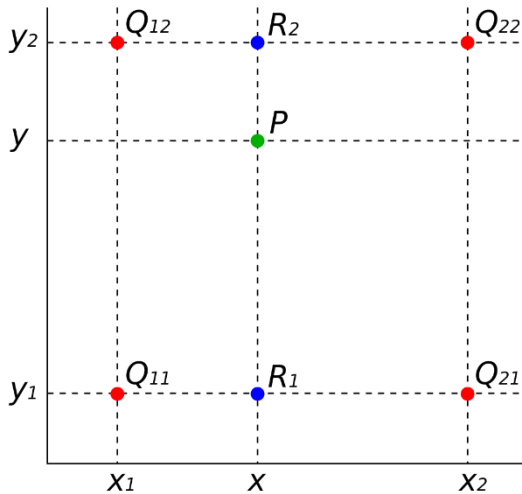
$$f(x) = \lambda_i y_i + \lambda_j y_j + \lambda_k y_k$$

Eigenschaften:

- $f(x)$ ist kontinuierlich
- Auf Dreiecksseiten nicht stetig differenzierbar
- lokaler Interpolator der nur (x_i, y_i) , (x_j, y_j) , (x_k, y_k) berücksichtigt, wenn $x \in \mathcal{C}(x_i, x_j, x_k)$
- Extrapolation ist konzeptuell **NICHT** möglich



Bilineare Interpolation auf einem Punktgitter



Seien die Messwerte $y_{i,j}$ auf einem Punktgitter $Q_{i,j} \in \mathbb{R}^2$ gegeben. Der unbekannte Wert an einem Punkt $P = (x^{(1)}, y^{(2)})$ innerhalb eines Gitterelements $Q_{i,j}, Q_{i+1,j}, Q_{i,j+1}, Q_{i+1,j+1}$ kann wie folgt vorhergesagt werden:

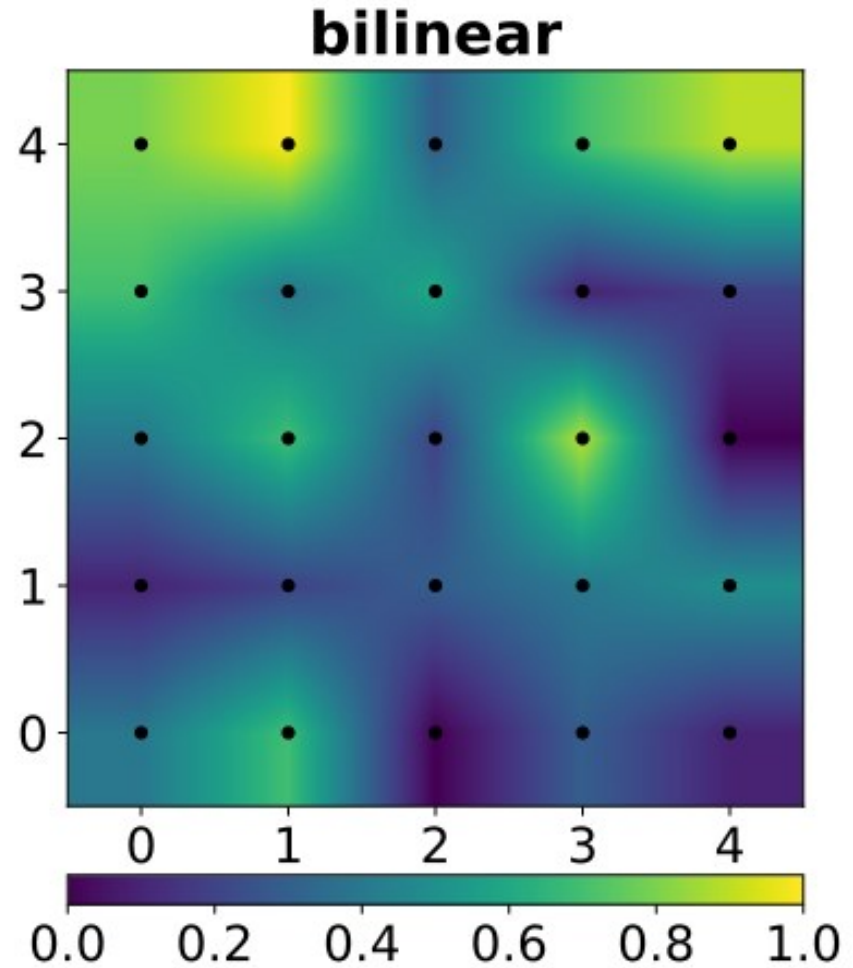
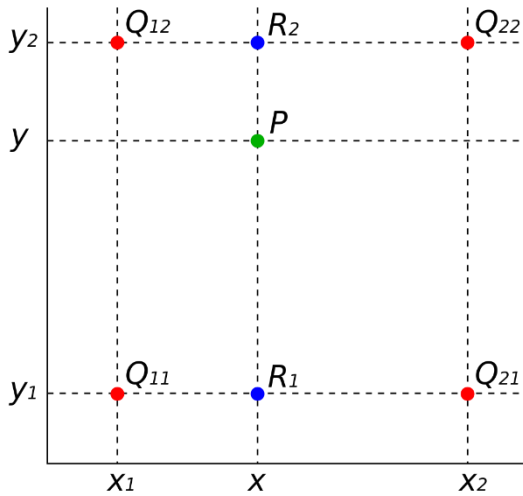
$$f(x^{(1)}, x^{(2)}) = \frac{x_{i,j+1}^{(2)} - x^{(2)}}{x_{i,j+1}^{(2)} - x_{i,j}^{(2)}} f_{i,j} + \frac{x^{(2)} - x_{i,j}^{(2)}}{x_{i,j+1}^{(2)} - x_{i,j}^{(2)}} f_{i,j+1}$$

mit

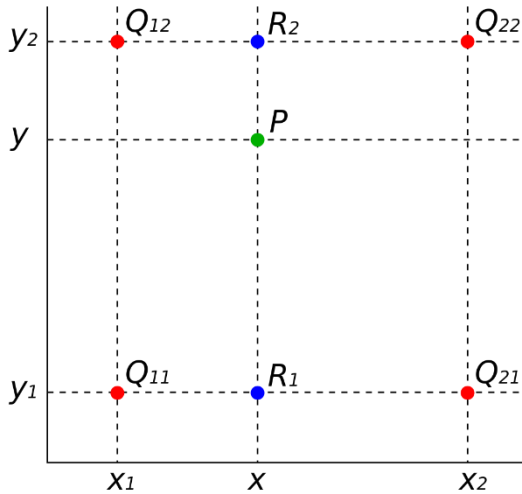
$$f_{i,j} = \frac{x_{i+1,j}^{(1)} - x^{(1)}}{x_{i+1,j}^{(1)} - x_{i,j}^{(1)}} y_{i,j} + \frac{x^{(1)} - x_{i,j}^{(1)}}{x_{i+1,j}^{(1)} - x_{i,j}^{(1)}} y_{i+1,j}$$

$$f_{i,j+1} = \frac{x_{i+1,j}^{(1)} - x^{(1)}}{x_{i+1,j}^{(1)} - x_{i,j}^{(1)}} y_{i,j+1} + \frac{x^{(1)} - x_{i,j}^{(1)}}{x_{i+1,j}^{(1)} - x_{i,j}^{(1)}} y_{i+1,j+1}$$

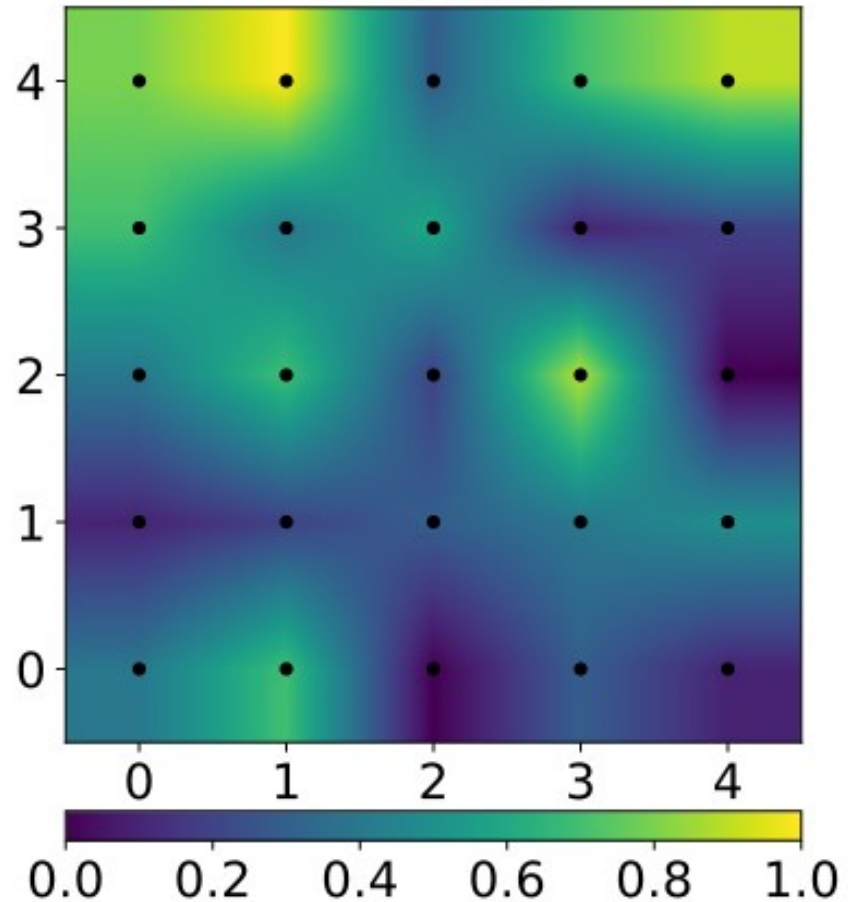
Bilinear Interpolation



Bilinear Interpolation



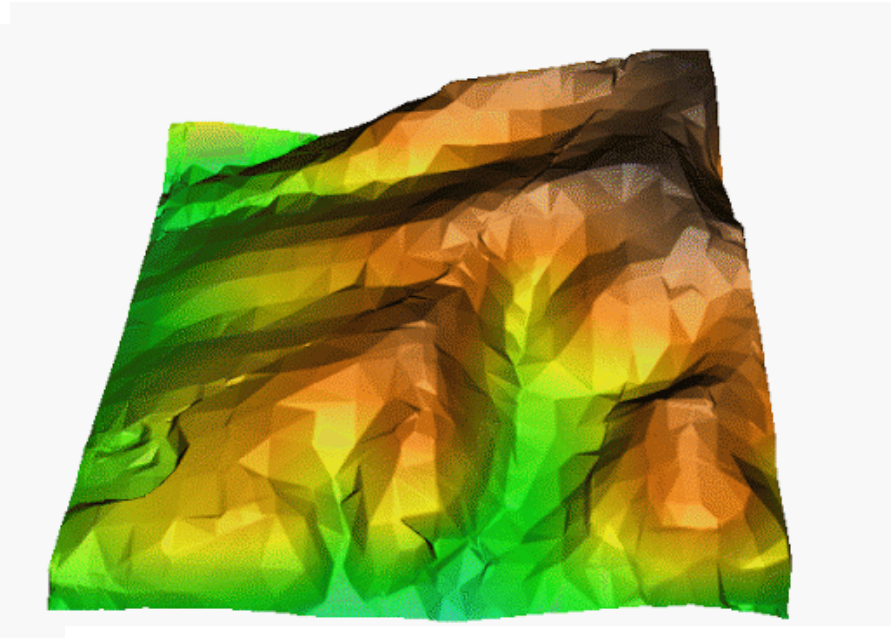
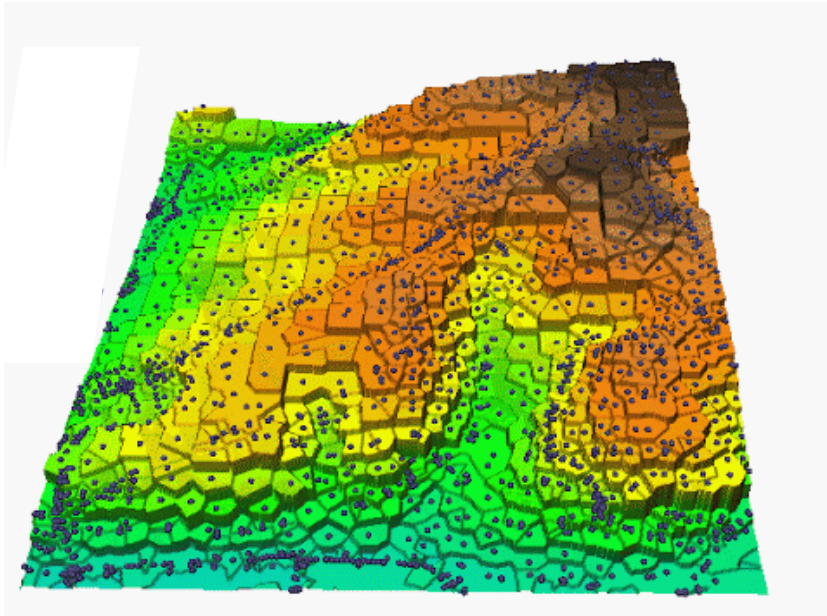
bilinear



Eigenschaften:

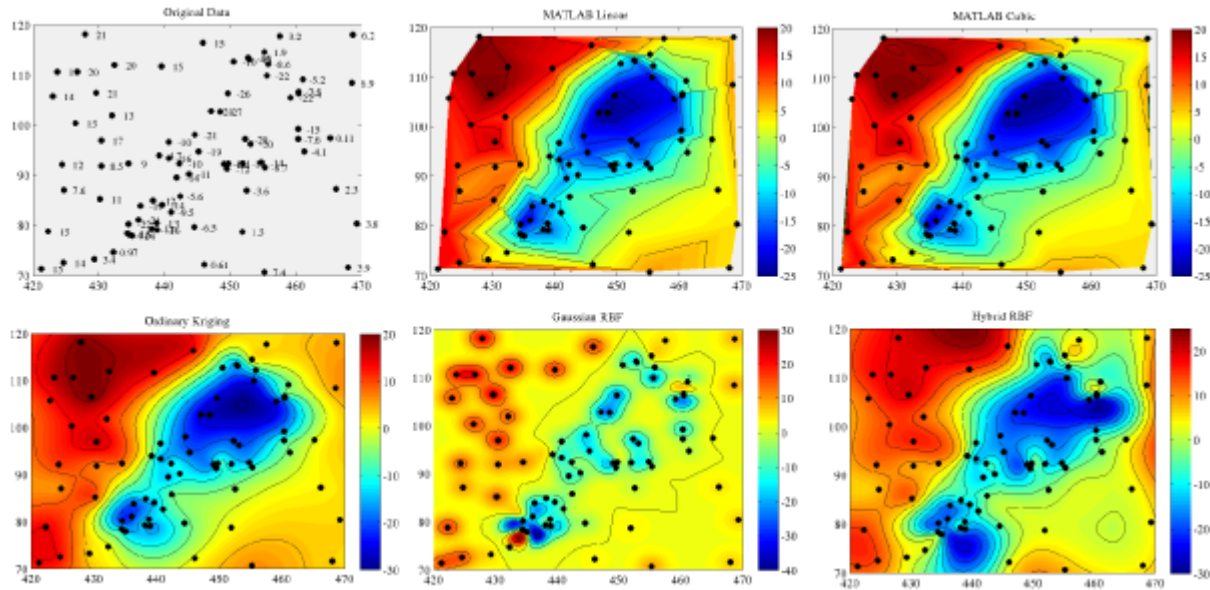
- $f(x)$ ist kontinuierlich
- auf Gitterkanten nicht differenzierbar
- Lokaler Interpolator, der sich nur auf $(x_{i,j}, y_{i,j}), (x_{i+1,j}, y_{i+1,j}), (x_{i,j+1}, y_{i,j+1}), (x_{i+1,j+1}, y_{i+1,j+1})$ bezieht, wenn x innerhalb dieses Gitterelements ist
- Extrapolation ist konzeptionell nicht möglich

Nearest Neighbour vs Lin. Interp. auf Dreiecken



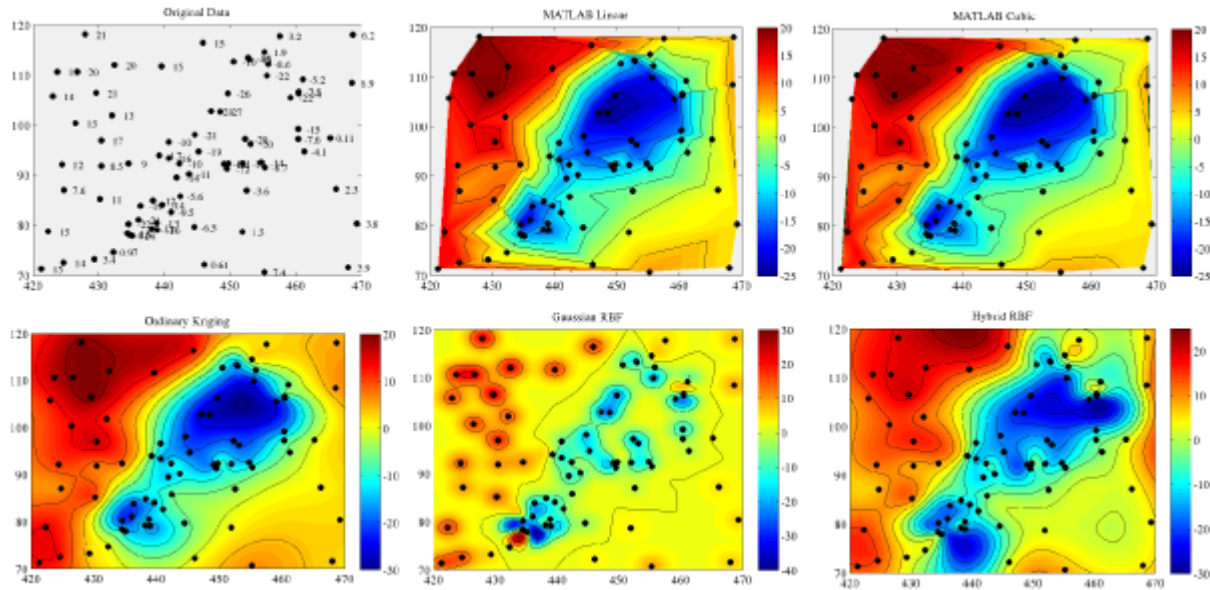
Gitter-freie Interpolation

- Inverse distance weighting (IDW)
- Polynom Interpolation
- Splines und Radial Basis Functions



Gitter-freie Interpolation

- Inverse distance weighting (IDW)
- ~~Polynom Interpolation~~ Polynom Approximation
- Splines und Radial Basis Functions



Inverse Distance Weighting (IDW)

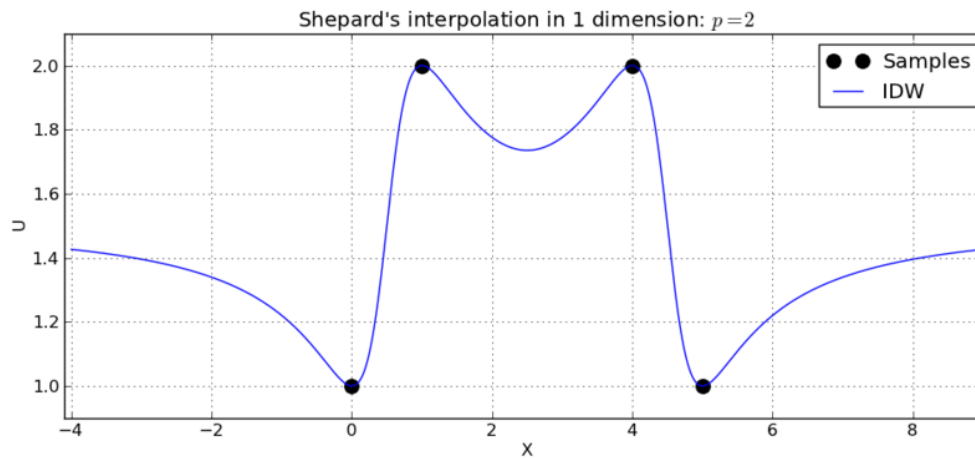
Annahme: Der Wert an einem beliebigen Punkt hängt nur vom Abstand zu den Datenpunkten ab.

Gegeben seien Tupel $(x_i, y_i); i = 1, \dots, N; x_i \in \mathbb{R}^n$. Der unbekannte Wert $f(x)$ an einem beliebigen Punkt $x \in \mathbb{R}^n$ lässt sich vorhersagen mit:

$$f(x) = \frac{\sum_{i=1}^N \lambda_i(x) y_i}{\sum_{i=1}^N \lambda_i(x)}$$

Mit

$$\lambda_i(x) = \frac{1}{|x - x_i|^p}$$



Inverse Distance Weighting (IDW)

Annahme: Der Wert an einem beliebigen Punkt hängt nur vom Abstand zu den Datenpunkten ab.

Gegeben seien Tupel $(x_i, y_i); i = 1, \dots, N; x_i \in \mathbb{R}^n$. Der unbekannte Wert $f(x)$ an einem beliebigen Punkt $x \in \mathbb{R}^n$ lässt sich vorhersagen mit:

$$f(x) = \frac{\sum_{i=1}^N \lambda_i(x) y_i}{\sum_{i=1}^N \lambda_i(x)}$$

Mit

$$\lambda_i(x) = \frac{1}{|x - x_i|^p}$$

Oder alternativ:

$$f(x) = \sum_{i=1}^N \lambda_i(x) y_i \text{ mit } \lambda_i(x) = \frac{\frac{1}{|x-x_i|^p}}{\sum_{j=1}^N \frac{1}{|x-x_j|^p}} = \frac{1}{1 + \sum_{j=1, j \neq i}^N \frac{|x-x_i|^p}{|x-x_j|^p}}$$

zum Vermeiden des globalen Vorfaktors ...

Inverse Distance Weighting (IDW)

$$f(x) = \sum_{i=1}^N \lambda_i(x) y_i \text{ mit } \lambda_i(x) = \frac{\frac{1}{|x-x_i|^p}}{\sum_{j=1}^N \frac{1}{|x-x_j|^p}} = \frac{1}{1 + \sum_{j=1, j \neq i}^N \frac{|x-x_i|^p}{|x-x_j|^p}}$$

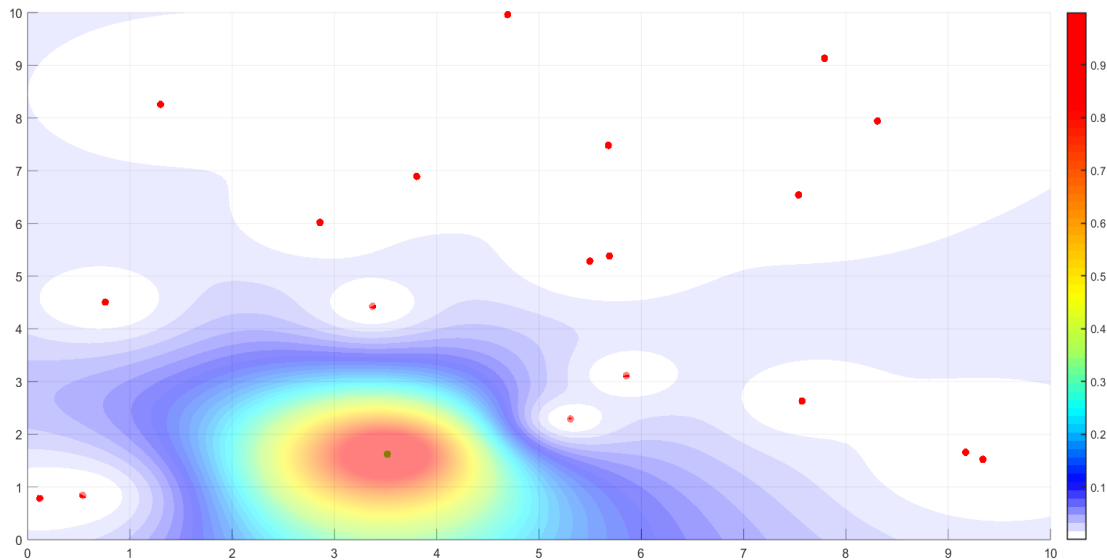


Illustration of weight function λ_1 for various $x \in [0, 1]^2$ with parameter $k = 2$.

Inverse Distance Weighting (IDW)

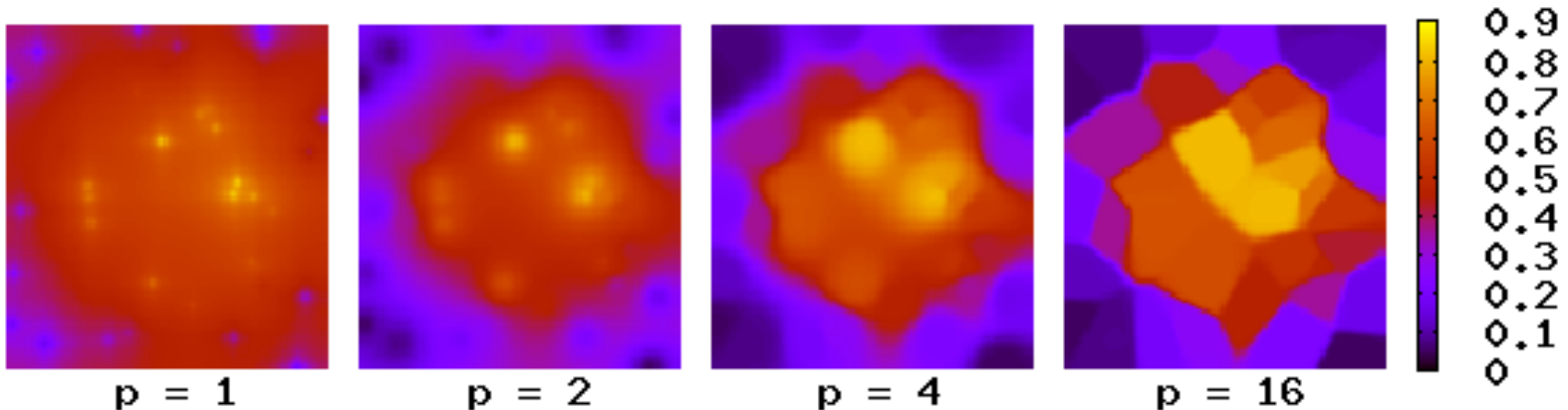
Gegeben seien Tupel $(x_i, y_i); i = 1, \dots, N; x_i \in \mathbb{R}^n$. Der unbekannte Wert $f(x)$ an einem beliebigen Punkt $x \in \mathbb{R}^n$ lässt sich vorhersagen mit:

$$f(x) = \frac{\sum_{i=1}^N \lambda_i(x) y_i}{\sum_{i=1}^N \lambda_i(x)}$$

Mit

$$\lambda_i(x) = \frac{1}{|x - x_i|^p}$$

Einfluss des Exponenten p als Lokalisierungsparameter:



Inverse Distance Weighting (IDW)

Gegeben seien Tupel $(x_i, y_i); i = 1, \dots, N; x_i \in \mathbb{R}^n$. Der unbekannte Wert $f(x)$ an einem beliebigen Punkt $x \in \mathbb{R}^n$ lässt sich vorhersagen mit:

$$f(x) = \frac{\sum_{i=1}^N \lambda_i(x) y_i}{\sum_{i=1}^N \lambda_i(x)}$$

Mit

$$\lambda_i(x) = \frac{1}{|x - x_i|^p}$$

Eigenschaften:

- $f(x)$ ist kontinuierlich;
- Globaler Interpolator, welcher alle Daten berücksichtigt;
- Gewichte hängen rein Punktabstand zw. Interpolations- und Datenpunkten ab;
- Extrapolation ist konzeptionell möglich, aber für

$$|x - x_i| \rightarrow \infty \forall i \Rightarrow f(x) \rightarrow \frac{\sum_{i=1}^N y_i}{N}$$

Polynom Interpolation

Annahme: Der unbekannte Wert an einem beliebigen Punkt lässt sich über eine Polynomfunktion $f(x) = \sum_{i=0}^m \lambda_i x^i$ vorhersagen.

Gegeben seien Tupel $(x_i, y_i); i = 1, \dots, N; x_i \in \mathbb{R}^n$. Sei $f(x) = \sum_{i=0}^m \lambda_i x^i$ ein Polynom vom Grad $m = N - 1$, dann kann man die Koeffizienten $\lambda_0, \dots, \lambda_{N-1} \in \mathbb{R}$ so wählen, dass

$$f(x_i) = y_i.$$

Um die Koeffizienten $\lambda_0, \dots, \lambda_m$ zu finden, muss folgendes Gleichungssystem gelöst werden:

$$\mathbf{M}\boldsymbol{\lambda} = \mathbf{y}$$

mit $\boldsymbol{\lambda} = (\lambda_0, \dots, \lambda_m)^T$, $\mathbf{y} = (y_1, \dots, y_N)$ und

$$\mathbf{M} = \begin{pmatrix} 1 & x_1 & x_1^2 & \cdots & x_1^m \\ 1 & x_2 & x_2^2 & \cdots & x_2^m \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_N & x_N^2 & \cdots & x_N^m \end{pmatrix}$$

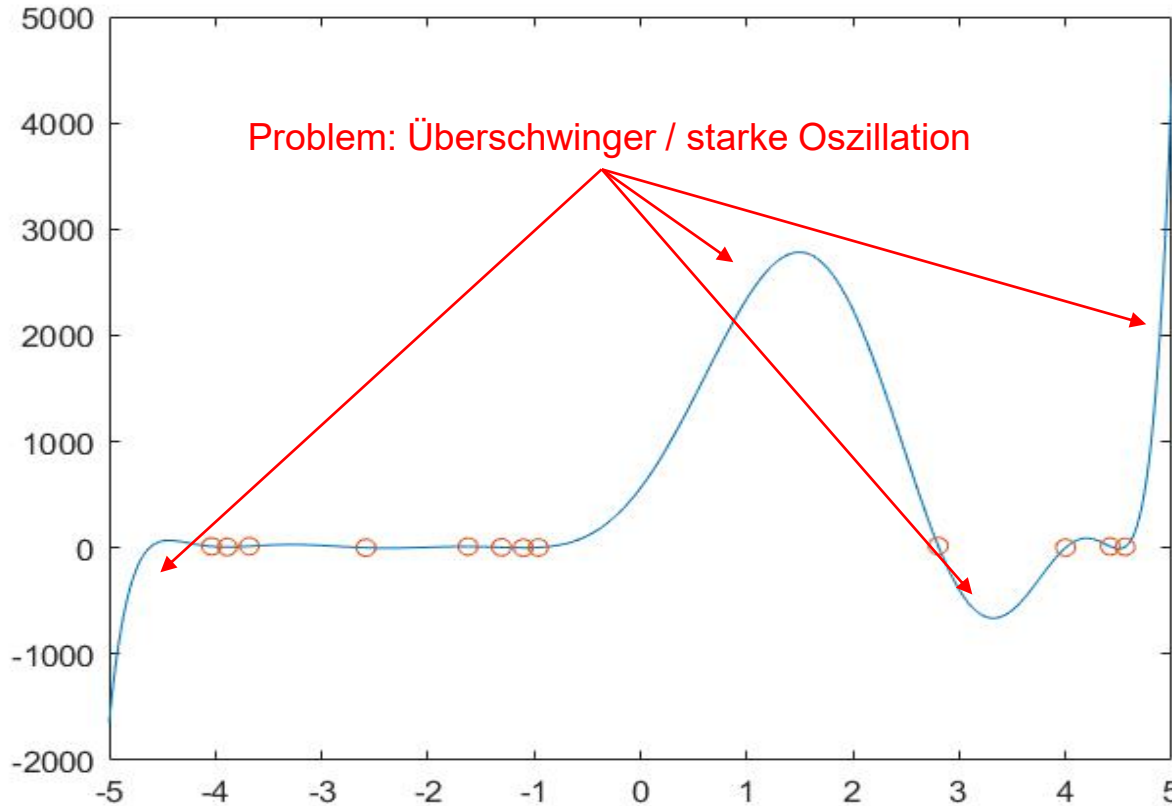
Polynom Interpolation

Annahme:
Polynom

Gegeben:
Grad

Um die

mit λ



en:

Interpolationspolynom für $N = 12$ Datenpunkte und Grad $m = N - 1 = 11$.

Polynom Approximation

Annahme: Der unbekannte Wert an einem beliebigen Punkt lässt sich über eine Polynomfunktion $f(x) = \sum_{i=0}^m \lambda_i x^i$ vorhersagen.

Gegeben seien Tupel $(x_i, y_i); i = 1, \dots, N; x_i \in \mathbb{R}^n$ und sei $f(x) = \sum_{i=0}^m \lambda_i x^i$ ein Polynom vom Grad m . Die Oszillation des Polynoms außerhalb der Daten kann dadurch reduziert werden, wenn der Grad $m \ll N - 1$ gewählt wird. Die Koeffizienten können nur noch so gewählt werden, dass sie

$$\min_{\lambda_0, \dots, \lambda_m} \sum_{i=1}^N |f(x_i) - y_i|^2$$

minimieren. Dieses Verfahren wird als **Approximation** (nicht-exakte Interpolation) bezeichnet, weil im Allgemeinen $f(x_i) = y_i \forall i$ **NICHT** erreicht werden kann. Es gilt nur noch

$$f(x_i) \approx y_i \forall i.$$

Eigenschaften:

- $f(x)$ ist ein Polynom vom Grad m (d.h. kontinuierlich und m -mal stetig differenzierbar)
- globaler Interpolator der **alle** (x_i, y_i) mit $1 \leq i \leq N$ berücksichtigt
- für $m \ll N - 1$: Approximations-Polynom $f(x_i) \approx y_i$ mit $1 \leq i \leq N$
- Extrapolation ist konzeptionell möglich

Splines 1-D

Ein **Spline von Grad n** ist eine Funktion, die für eine gegebene Menge Knoten $P = \{x_1, \dots, x_N\} \subset \mathbb{R}$ über jedem offenen Intervall (x_i, x_{i+1}) , (x_n, ∞) und $(-\infty, x_1)$ einem Polynom vom Grad n entspricht und deren Ableitungen auf den Knoten x_1, \dots, x_N bis zum Grad $n - 1$ stetig sind.

Der Raum aller Funktionen, die diesen Spline-Eigenschaften entsprechen, wird als $SP_{P,n}$ bezeichnet.

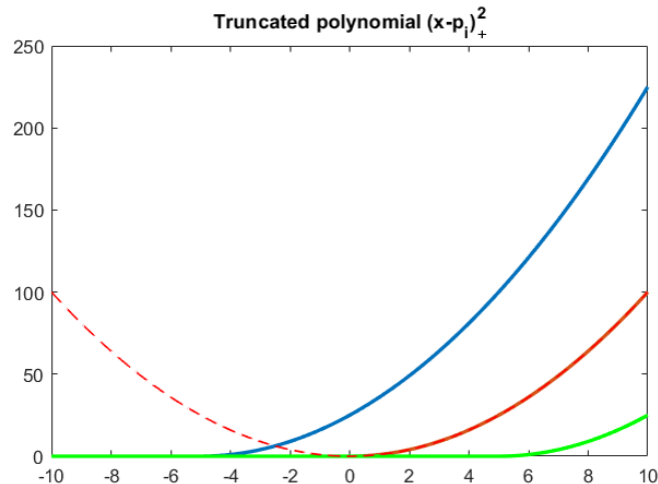
Splines 1-D

Ein **Spline von Grad n** ist eine Funktion, die für eine gegebene Menge Knoten $P = \{x_1, \dots, x_N\} \subset \mathbb{R}$ über jedem offenen Intervall (x_i, x_{i+1}) , (x_n, ∞) und $(-\infty, x_1)$ einem Polynom vom Grad n entspricht und deren Ableitungen auf den Knoten x_1, \dots, x_N bis zum Grad $n - 1$ stetig sind.

Der Raum aller Funktionen, die diesen Spline-Eigenschaften entsprechen, wird als $SP_{P,n}$ bezeichnet.

Ein Beispiel für die Element von $SP_{P,n}$ sind so genannte *verschränkte (truncated) Polynome*:

$$(x - p_i)_+^n = \begin{cases} (x - p_i)^n, & \text{falls } x \geq p_i, \\ 0, & \text{sonst} \end{cases}$$



Splines 1-D

Ein **Spline von Grad n** ist eine Funktion, die für eine gegebene Menge Knoten $P = \{x_1, \dots, x_N\} \subset \mathbb{R}$ über jedem offenen Intervall (x_i, x_{i+1}) , (x_n, ∞) und $(-\infty, x_1)$ einem Polynom vom Grad n entspricht und deren Ableitungen auf den Knoten x_1, \dots, x_N bis zum Grad $n - 1$ stetig sind.

Der Raum aller Funktionen, die diesen Spline-Eigenschaften entsprechen, wird als $SP_{P,n}$ bezeichnet.

Ein Beispiel für die Element von $SP_{P,n}$ sind so genannte *verschränkte (truncated)* Polynome:

$$(x - p_i)_+^n = \begin{cases} (x - p_i)^n, & \text{falls } x \geq p_i, \\ 0, & \text{sonst} \end{cases}$$

Die n -te Ableitung $S^{(n)}$ eines Splines aus $SP_{P,n}$ führt zu einer Stufenfunktion mit Sprüngen an x_i .

Splines 1-D

Die Dimension von $SP_{P,n}$ ist

$$\dim(SP_{P,n}) = n + N + 1,$$

dies kann wie folgt interpretiert werden: Auf jedem Intervall (x_i, x_{i+1}) liegen $n + 1$ Freiheitsgrade. Durch die Ableitungsbedingung an den Knoten geht dort ein n Freiheitsgrade verloren:

$$\dim(SP_{P,n}) = (N + 1)(n + 1) - nN = n + N + 1$$

Splines 1-D

Die Dimension von $SP_{p,n}$ ist

$$\dim(SP_{p,n}) = n + N + 1,$$

dies kann wie folgt interpretiert werden: Auf jedem Intervall (x_i, x_{i+1}) liegen $n + 1$ Freiheitsgrade. Durch die Ableitungsbedingung an den Knoten geht dort ein n Freiheitsgrade verloren:

$$\dim(SP_{p,n}) = (N + 1)(n + 1) - nN = n + N + 1$$

Ein Spline S kann daher wie folgt ausgedrückt werden:

$$S(x) = \sum_{i=1}^N \lambda_i (x - x_i)_+^n + \sum_{j=0}^n \lambda_{j+N+1} x^j$$

Die Gewichte $\lambda_1, \dots, \lambda_{n+N+1}$ werden über ein lineares Gleichungssystem bestimmt, ähnlich der Polynom-Interpolation / -approximation,

Splines 1-D

Die Dimension von $SP_{P,n}$ ist

$$\dim(SP_{P,n}) = n + N + 1,$$

dies kann wie folgt interpretiert werden: Auf jedem Intervall (x_i, x_{i+1}) liegen $n + 1$ Freiheitsgrade. Durch die Ableitungsbedingung an den Knoten geht dort ein n Freiheitsgrade verloren:

$$\dim(SP_{P,n}) = (N + 1)(n + 1) - nN = n + N + 1$$

Ein Spline S kann daher wie folgt ausgedrückt werden:

$$S(x) = \sum_{i=1}^N \lambda_i (x - x_i)_+^n + \sum_{j=0}^n \lambda_{j+N+1} x^j$$

Die Gewichte $\lambda_1, \dots, \lambda_{n+N+1}$ werden über ein lineares Gleichungssystem bestimmt, ähnlich der Polynom-Interpolation / -approximation.

Zumeist liegen nur ein Messwert y_i pro Knoten x_i und damit nur N Randbedingungen vor. Um das o.g. Problem lösen zu können, werden $n + 1$ zusätzliche Randbedingungen benötigt. Im Fall von $n = 2m - 1$ könnten diese z.B. sein:

$$S^{(k)}(x_1) = S^{(k)}(x_N) = 0, k = m, \dots, n - 1$$

Splines, die diese Bedingung erfüllen werden als **natürliche Splines** bezeichnet, der zugehörige Raum $SPN_{P,n}$.

Warum sind Splines so nützlich?

Liegen die Werte y_i an den Punkten x_i mit $i = 1, \dots, N$ vor, so gilt für den Spline $S \in \text{SPN}_{P,n}$, der

$$S(x_i) = y_i, i = 1, \dots, N$$

interpoliert, dass er folgenden Ausdruck minimiert:

$$\min_{f \text{ interpolates } P} \int_{x_i}^{x_N} |f^n(x)|^2 dx$$

(Minimierung der n-ten Ableitung der Interpolationsfunktion).

Diese Splines werden auch als **kubische Splines** bezeichnet und kombinieren den einfachen Charakter von Polynominterpolation mit zusätzlicher Stabilität und geringerer Oszillation.

Warum sind Splines so nützlich?

Liegen die Werte y_i an den Punkten x_i mit $i = 1, \dots, N$ vor, so gilt für den Spline $S \in \text{SPN}_{P,n}$, der

$$S(x_i) = y_i, i = 1, \dots, N$$

interpoliert, dass er folgenden Ausdruck minimiert:

$$\min_{f \text{ interpolates } P} \int_{x_i}^{x_N} |f^n(x)|^2 dx$$

(Minimierung der n-ten Ableitung der Interpolationsfunktion).

Diese Splines werden auch als **kubische Splines** bezeichnet und kombinieren den einfachen Charakter von Polynominterpolation mit zusätzlicher Stabilität und geringerer Oszillation.

Eigenschaften:

- $f(x)$ ist kontinuierlich und min. $n - 1$ -mal stetig differenzierbar
- Globaler Interpolator
- Extrapolation ist konzeptuell möglich (hier können dann aber Oszillationen auftreten)