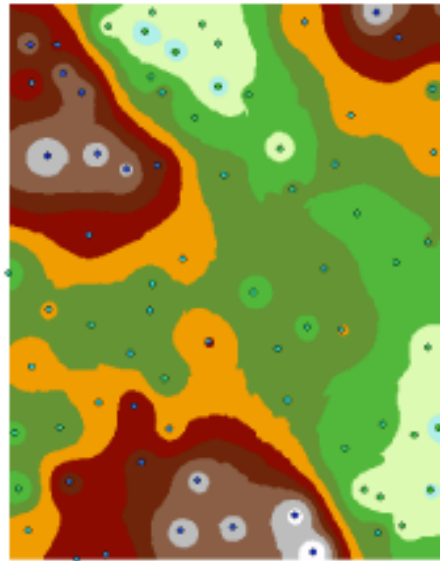
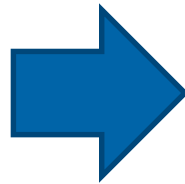
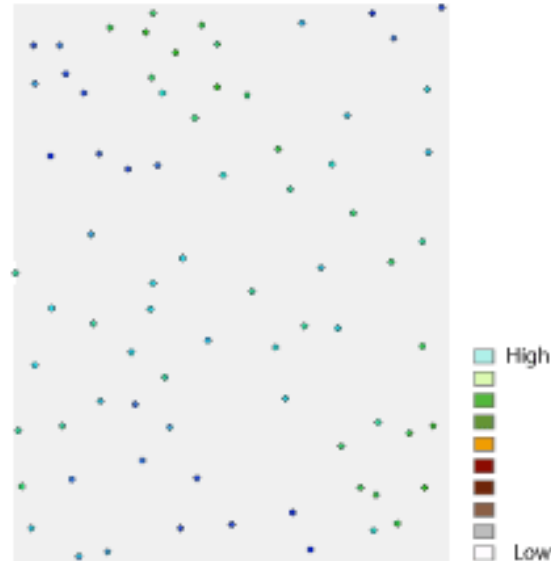


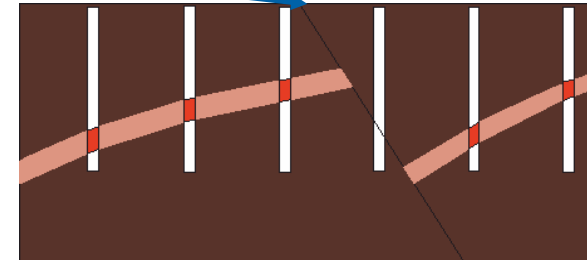
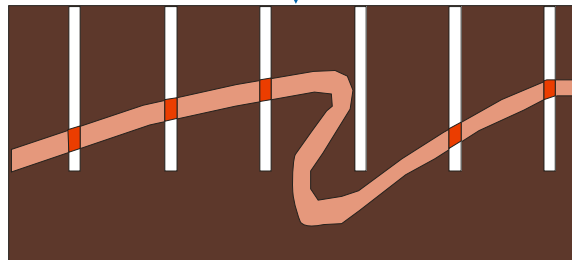
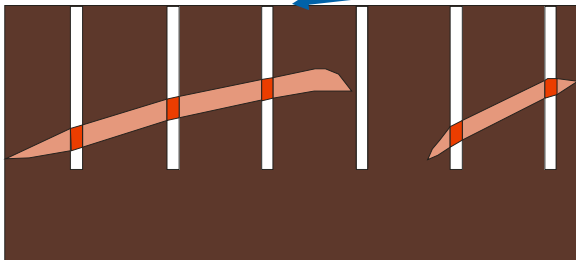
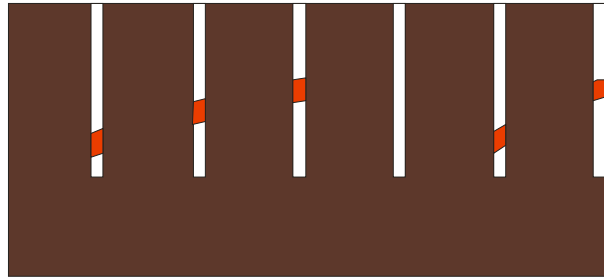
3D Geomodellierung

3. Interpolation



$$y_j = \sum_i w_{ij} y_i$$

Hinter der Hacke ist es duster ...



Interpolation

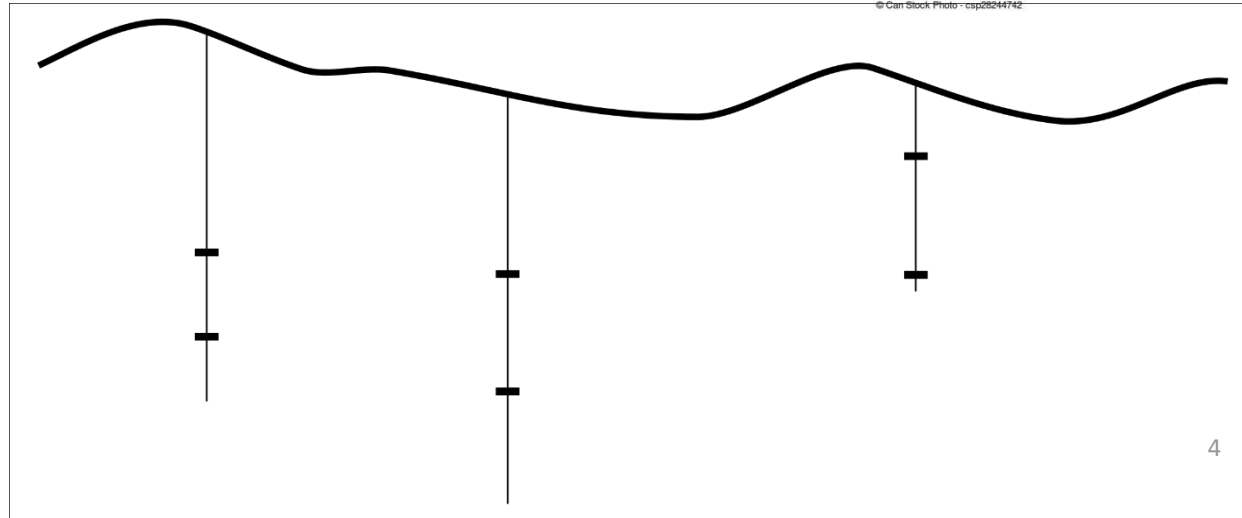
In der numerischen Mathematik bezeichnet der Begriff **Interpolation** (aus lateinisch *inter* = dazwischen und *polire* = glätten, schleifen) eine Klasse von Problemen und Verfahren. Zu gegebenen diskreten Daten (z. B. Messwerten) soll eine stetige Funktion (die sogenannte *Interpolante* oder *Interpolierende*) gefunden werden, die diese Daten abbildet. Man sagt dann, die Funktion *interpoliert* die Daten



[https://de.wikipedia.org/wiki/Interpolation_\(Mathematik\)](https://de.wikipedia.org/wiki/Interpolation_(Mathematik))

Interpolation in der Geomodellierung

Ausgehend von bekannten Daten soll eine Vorschrift gefunden werden, welche eine geowissenschaftlich plausible Vorhersage von unbekanntem Werten an Positionen ohne gegebene Werte erlaubt.

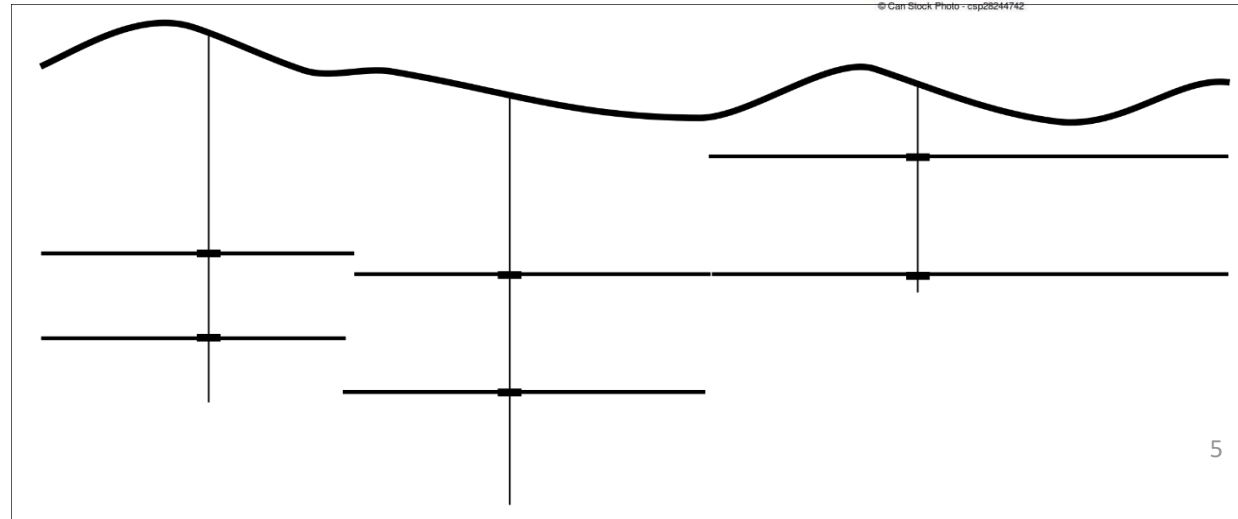


Interpolation in der Geomodellierung

Ausgehend von bekannten Daten soll eine Vorschrift gefunden werden, welche eine geowissenschaftlich plausible Vorhersage von unbekanntem Werten an Positionen ohne gegebene Werte erlaubt.



- Möglichst wenig Annahmen
- Mathematisch einfach

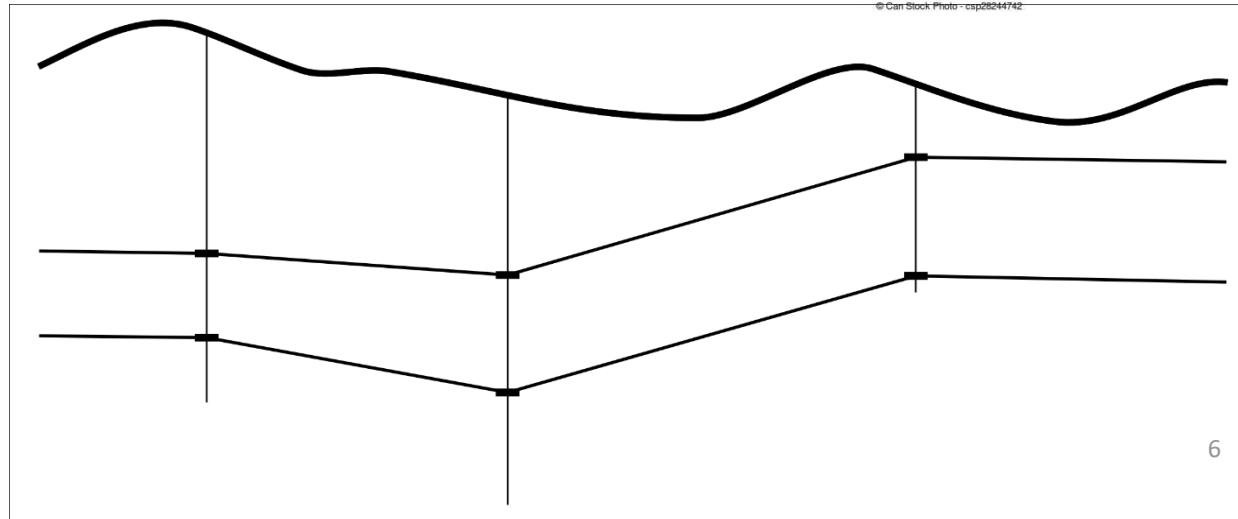


Interpolation in der Geomodellierung

Ausgehend von bekannten Daten soll eine Vorschrift gefunden werden, welche eine geowissenschaftlich plausible Vorhersage von unbekanntem Werten an Positionen ohne gegebene Werte erlaubt.



- Möglichst wenig Annahmen
- Mathematisch einfach

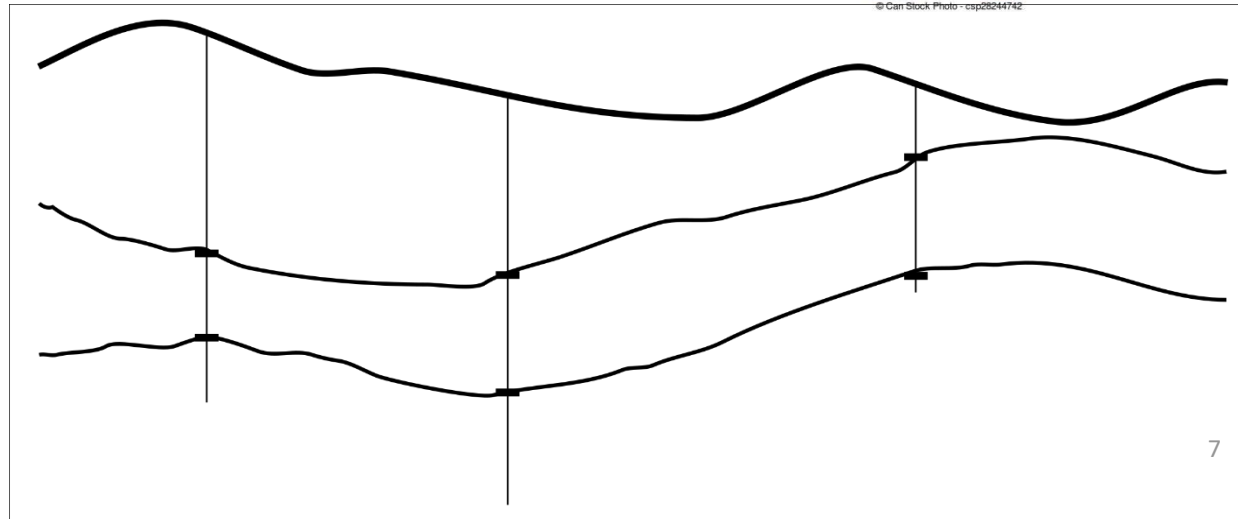


Interpolation in der Geomodellierung

Ausgehend von bekannten Daten soll eine Vorschrift gefunden werden, welche eine geowissenschaftlich plausible Vorhersage von unbekanntem Werten an Positionen ohne gegebene Werte erlaubt.



- Geologisch plausibel
- Mathematisch kompliziert

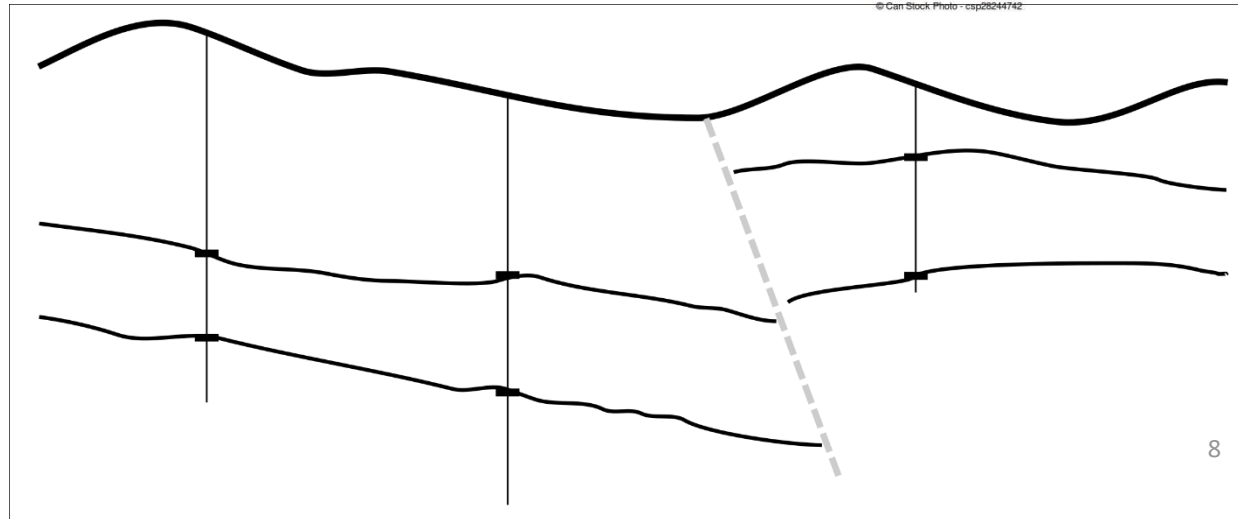


Interpolation in der Geomodellierung

Ausgehend von bekannten Daten soll eine Vorschrift gefunden werden, welche eine geowissenschaftlich plausible Vorhersage von unbekanntem Werten an Positionen ohne gegebene Werte erlaubt.



- Geologisch plausibel
- Mit Zusatzinformationen
- Mathematisch kompliziert



Mathematische Grundlagen

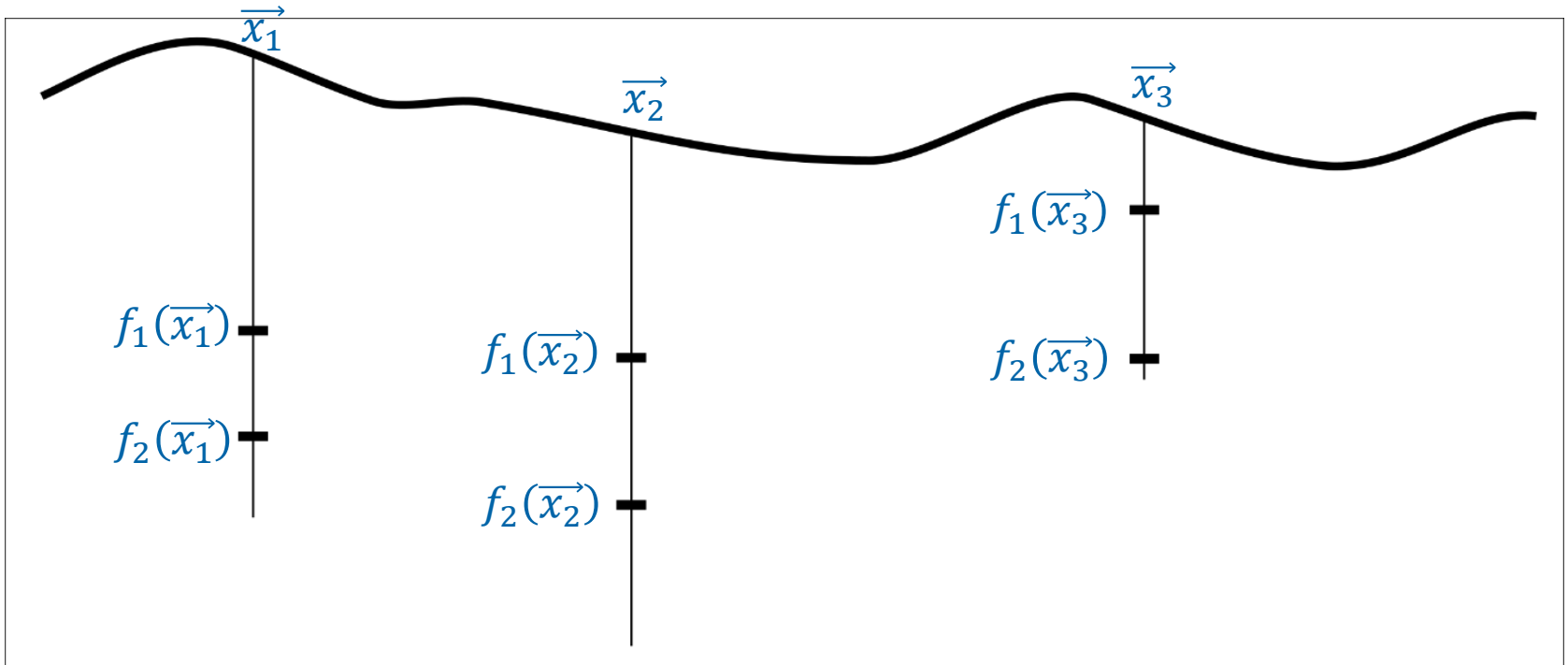
Ausgangslage:

- **Gegeben** sind n verschiedene Punkte $\vec{x}_i \in \mathbb{R}^p$ mit bekannten Messwerten f_i mit $(\vec{x}_i, f_i) \in D \subset \mathbb{R}^{p+1}, i = 1 \dots n$.

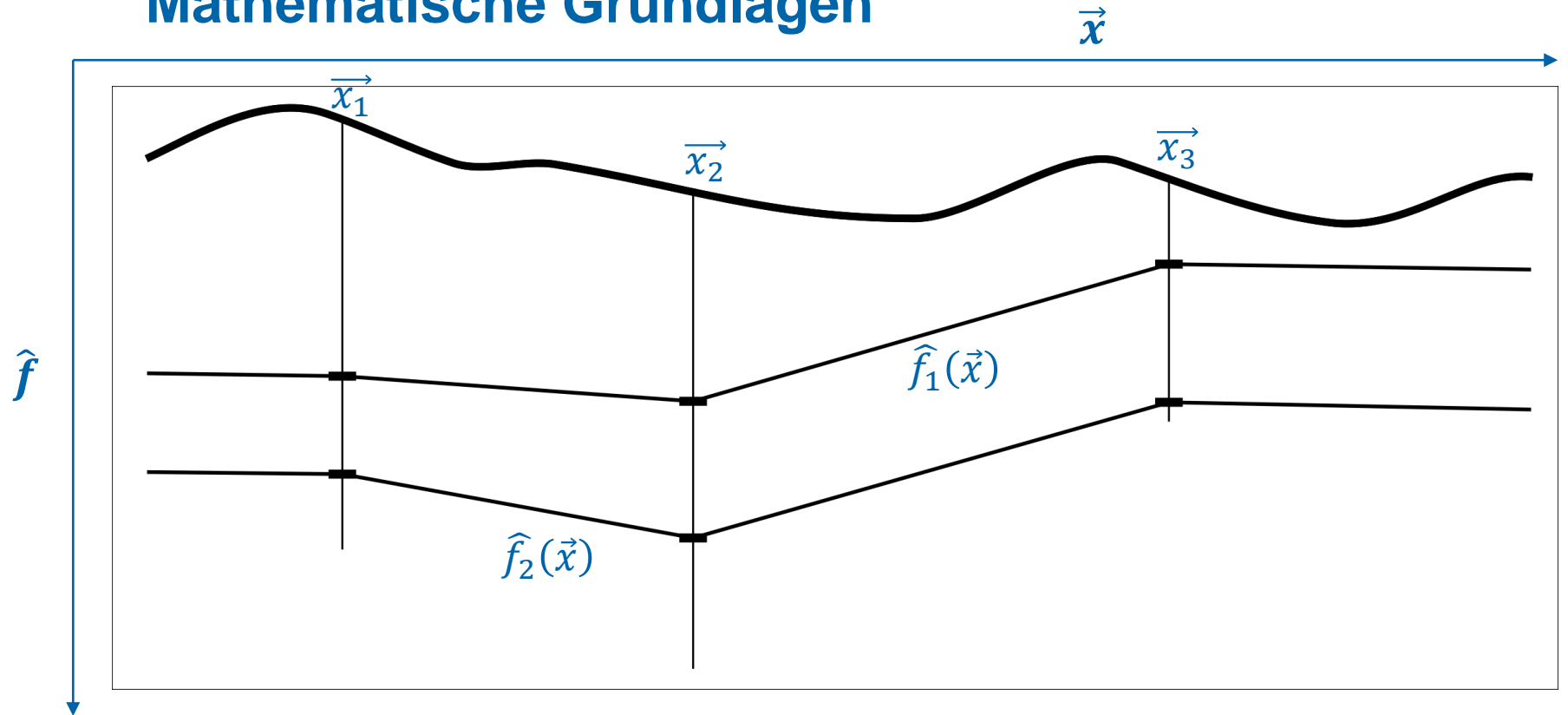
$f_i \in \mathbb{R}$ kann als $f(\vec{x}_i), \vec{x}_i \in \mathbb{R}^p$ aufgefasst werden.

- **Gesucht** wird eine Rechenvorschrift \hat{f} , welche jedem $\vec{x} \in D \subset \mathbb{R}^p$ einen eindeutigen Wert $\hat{f}(\vec{x})$ zuordnet. \hat{f} bildet dabei eine **Modellvorstellung** zur wahren, aber unbekanntem Natur \bar{f} des zu interpolierenden Sachverhalts ab.

Mathematische Grundlagen



Mathematische Grundlagen



Mathematische Grundlagen

(Exakte) Interpolation*:

$$\hat{f}(\vec{x}_i) = f(\vec{x}_i), i = 1 \dots n$$

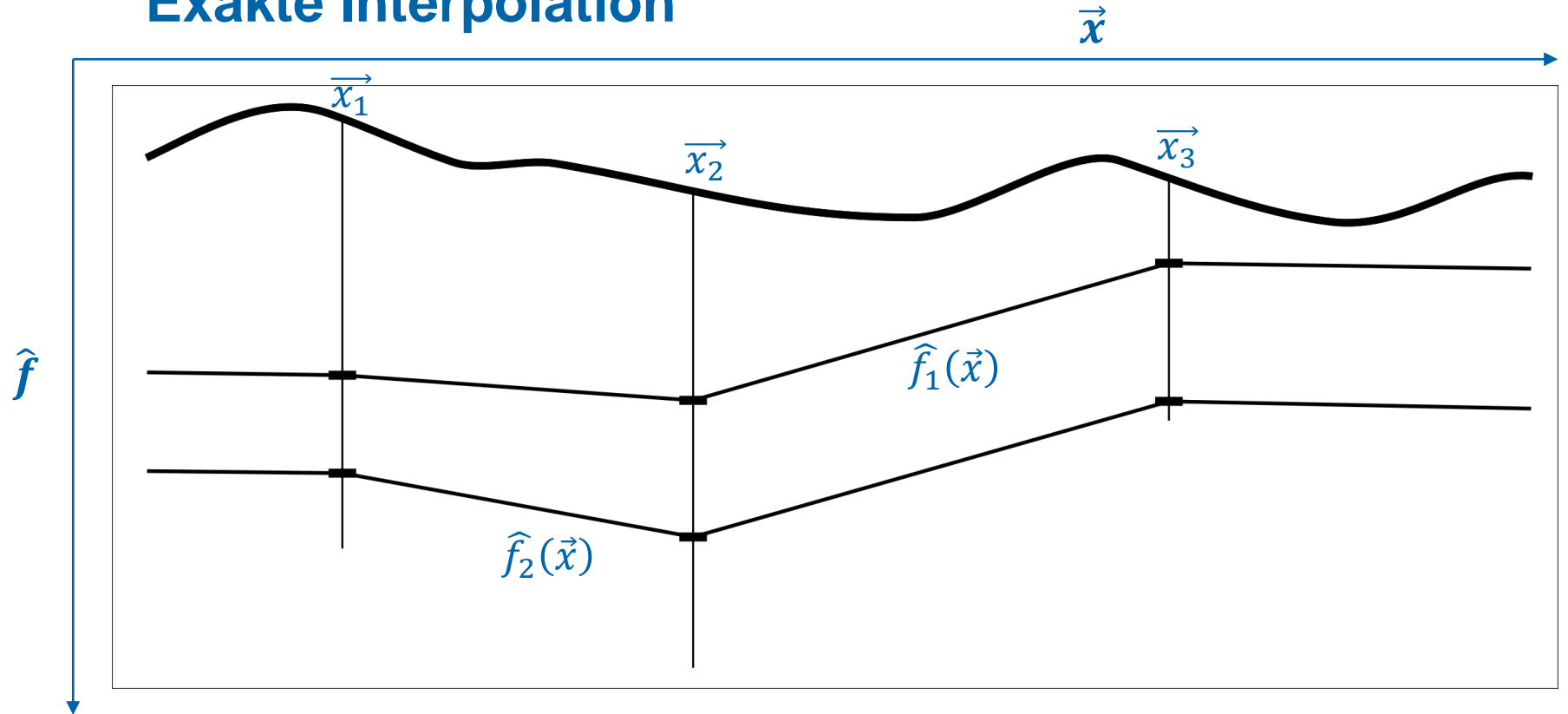
Nicht-exakte Interpolation / Approximation:

$$\hat{f}(\vec{x}_i) \sim f(\vec{x}_i), i = 1 \dots n$$

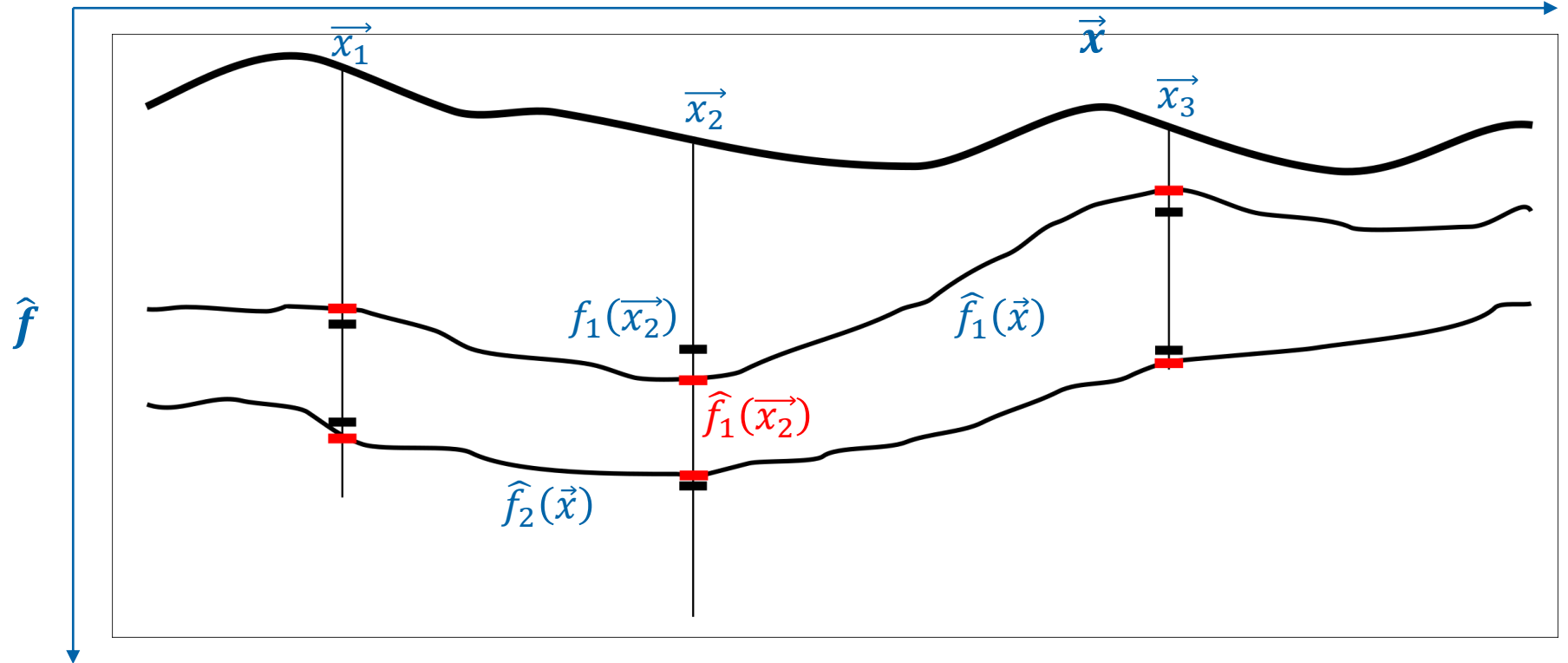
Wobei \sim noch spezifiziert werden muss. Hier wird davon ausgegangen, dass alle $f(\vec{x}_i)$ einen unbekanntem Fehler $\varepsilon \neq 0$ aufweisen.

$$f(\vec{x}_i) = \bar{f}(\vec{x}_i) + \varepsilon; \text{ mit } \bar{f}(\vec{x}_i) \rightarrow \text{wahrer Wert}$$

Exakte Interpolation



Nicht-exakte Interpolation / Approximation



Mathematische Grundlagen

Allgemeine Problemstellung für eine klassische Interpolation:

$$\hat{f}(\vec{x}) = \sum_i^n w_i(\vec{x}) f(\vec{x}_i)$$

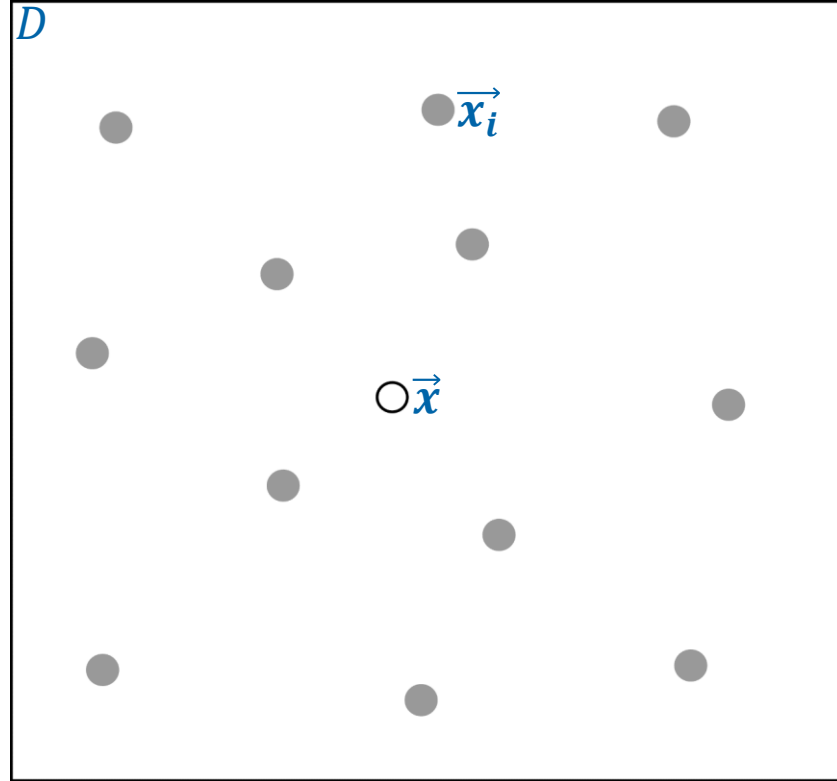


Gewichtetes Mittel über alle Datenpunkte \vec{x}_i am Punkt \vec{x} .

1. Teilproblem: Bestimmung der Gewichte $w_i(\vec{x})$.

Mathematische Grundlagen

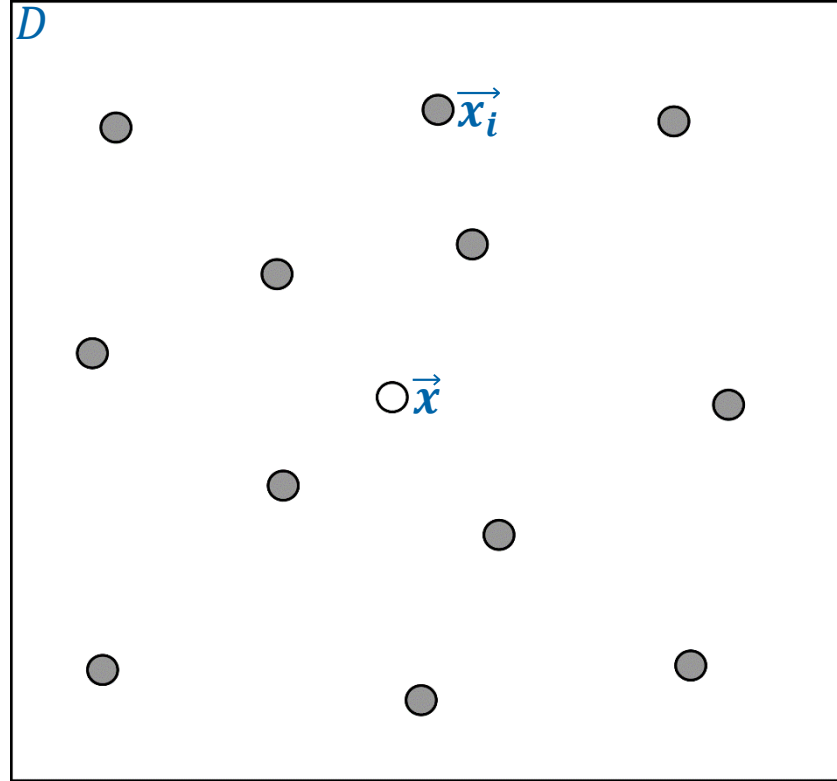
Lokale oder globale Interpolation:



Mathematische Grundlagen

Lokale oder globale Interpolation:

Global: $w_i(\vec{x}) \neq 0 \forall \vec{x}_i$

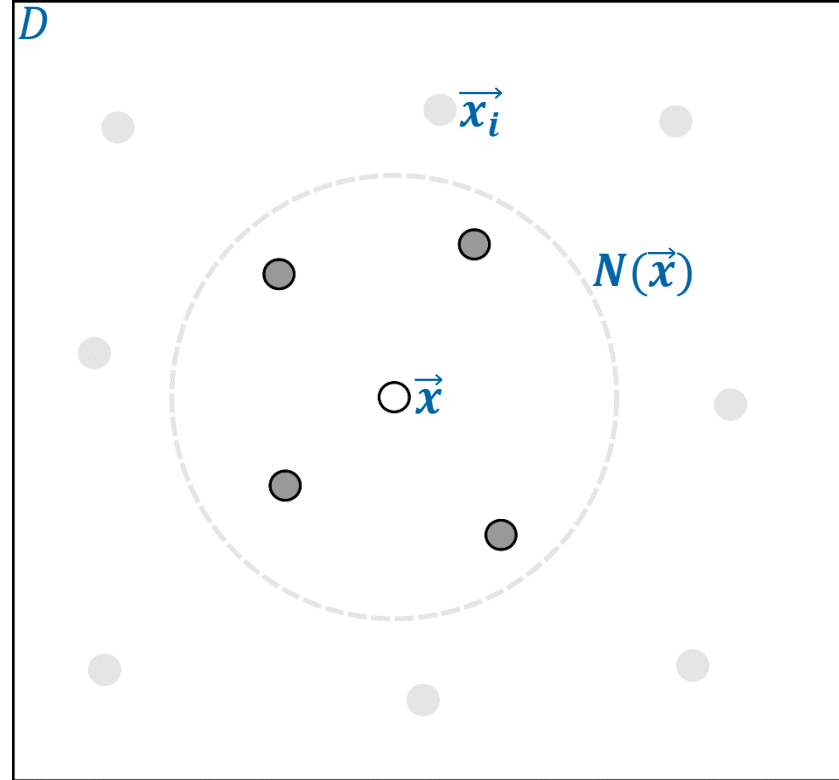


Mathematische Grundlagen

Lokale oder globale Interpolation:

Global: $w_i(\vec{x}) \neq 0 \forall \vec{x}_i$

Lokal: $w_i(\vec{x}) \begin{cases} \neq 0 \forall \vec{x}_i \in N(\vec{x}) \subset D \\ = 0 \forall \vec{x}_i \notin N(\vec{x}) \subset D \end{cases}$



2. Teilproblem: Bestimmung der Nachbarschaftsbeziehungen $N(\vec{x})$.

Interpolation vs Extrapolation

inter (lat.): zwischen

polire (lat.): glätten, schleifen

extra (lat.): außerhalb

Interpolation: Vorhersage zwischen gegebenen Datenpunkten.

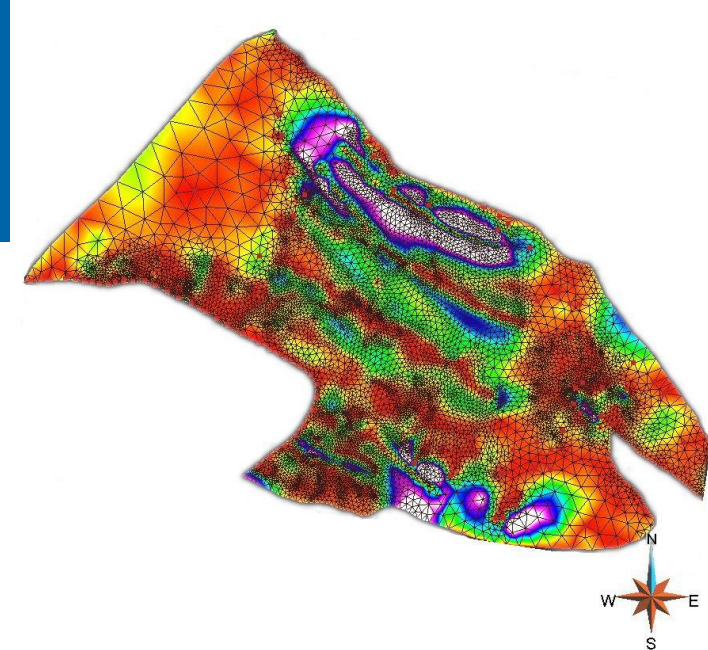
- Bezüglich des Modells plausibel

Extrapolation: Vorhersage außerhalb gegebenen Datenpunkten.

- Vorhersage ohne ausreichende Datengrundlage
- Häufig nicht plausibel
- Für bestimmte Verfahren nicht möglich

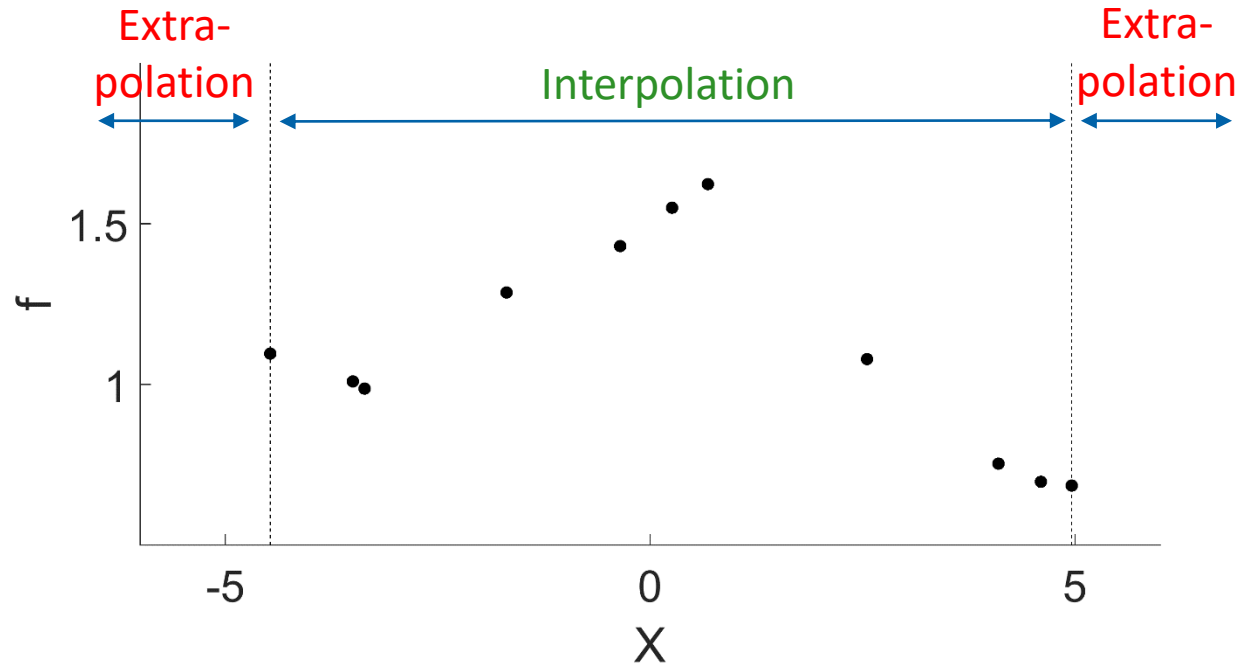
Take-home questions:

- 1) Was ist der Zweck von Interpolation?
- 2) Worin unterscheiden sich Interpolation und Approximation?
- 3) Worin unterscheiden sich globale von lokalen Interpolationsverfahren?
- 4) Warum muss Extrapolation als kritisch angesehen werden?



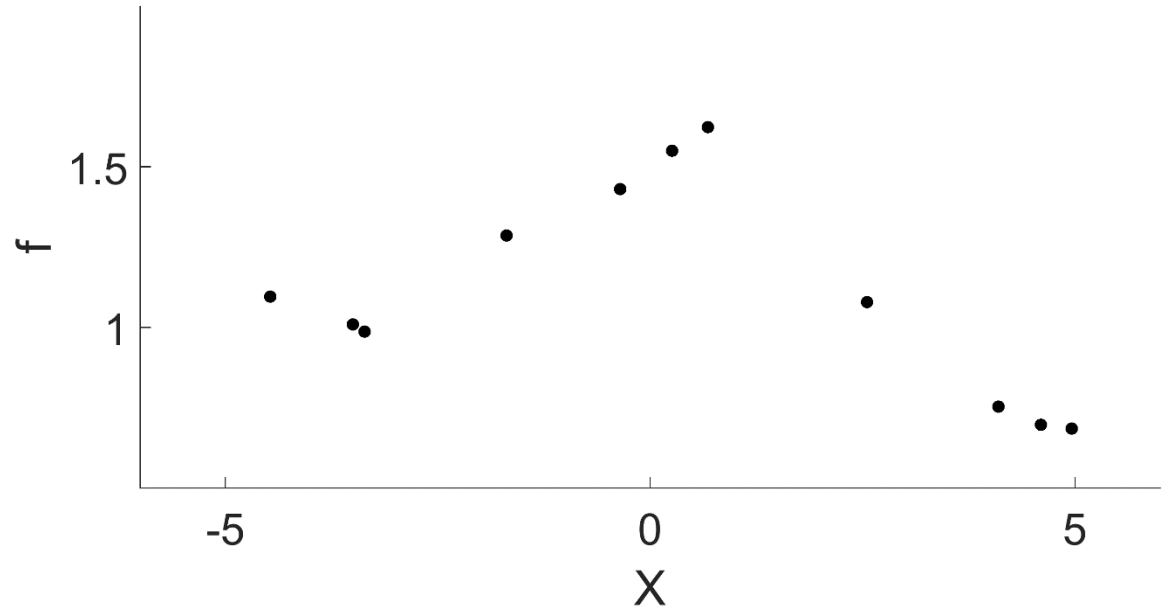
Institut für Geophysik und Geoinformatik
Dr. Peter Menzel
Gustav-Zeuner-Str. 12
09599 Freiberg
Tel. +49(0)3731 39-3815

Wo würde Interpolation und wo Extrapolation stattfinden?



Welche Modellannahme könnte diesen Daten zugrunde gelegt werden?

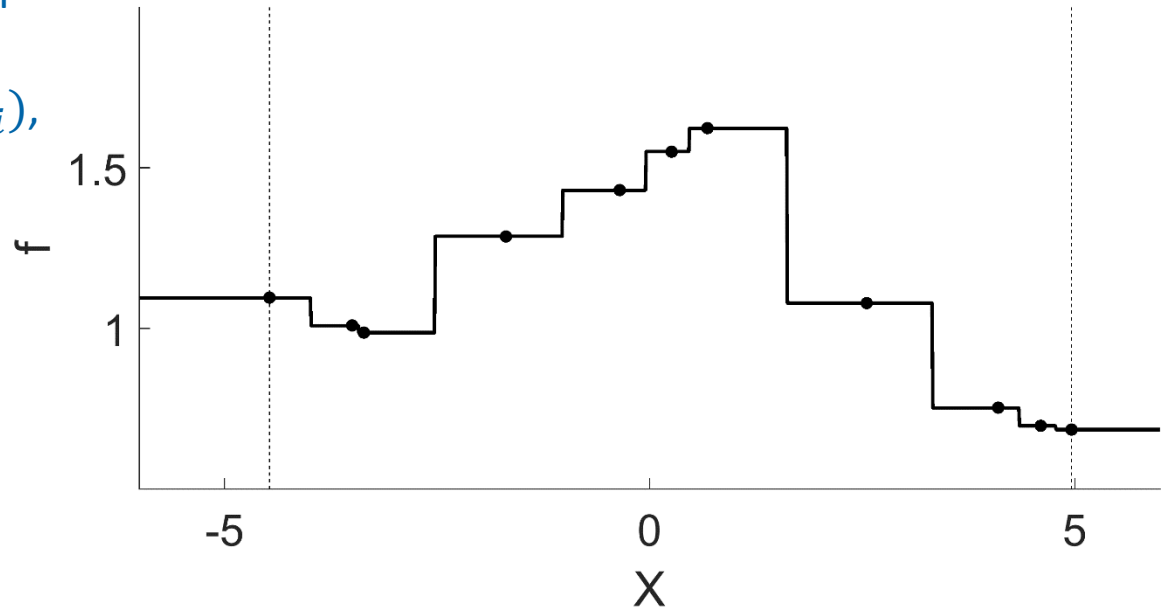
Modellannahme: ... ?



Stückweise konstante Interpolation

Modellannahme:

Ein vorhergesagter Wert für eine Position x entspricht dem bekannten Wert $f_i(x_i)$, der sich „am nächsten“ befindet.



Stückweise konstante Interpolation

1D Fall: $x_1 < x_2 < \dots < x_n$ mit $D = [x_1, x_n] \subset \mathbb{R}^1$

Mit $v_i = x_i + \frac{x_{i-1} - x_i}{2} = \frac{x_i + x_{i+1}}{2}, i = 1, \dots, n - 1$ kann man eine Zerlegung von D in Intervalle $D_1 = [x_1, v_1], D_i = (v_{i-1}, v_i], D_n = (v_{n-1}, x_n]$ definieren, so dass für jedes $x \in D_i$ der Abstand zu x_i kleiner ist als zu jedem anderen x_j .

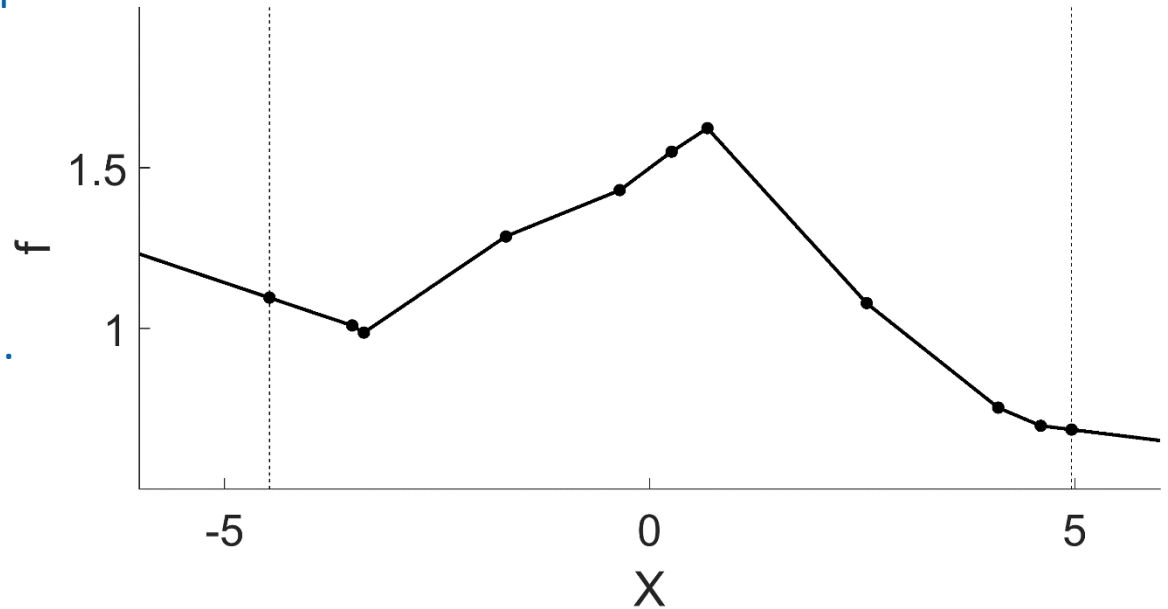
$$\hat{f}(x) = \sum_{i=1}^n \mathbb{I}_{D_i}(x) f_i \quad \text{mit} \quad \mathbb{I}_{D_j}(x) = \begin{cases} 1, & x \in D_j \\ 0, & x \notin D_j \end{cases}$$

Eine Verallgemeinerung für \mathbb{R}^n ist sehr einfach möglich (*nearest-neighbor-interpolant*).

Stückweise lineare Interpolation

Modellannahme:

Ein vorhergesagter Wert für eine Position x zwischen 2 bekannten benachbarten Positionen x_i und x_j ergibt sich aus dem Distanz-gewichteten Mittel zwischen $f_i(x_i)$ und $f_j(x_j)$.



Stückweise lineare Interpolation


Variante (a)

$\forall x_i, i = 2, \dots, n - 1$ Funktion w_i auf $[x_{i-1}, x_{i+1}]$ mit

$$w_i(x) = \begin{cases} \frac{x - x_{i-1}}{x_i - x_{i-1}}, & x \in [x_{i-1}, x_i] \\ \frac{x_{i+1} - x}{x_{i+1} - x_i}, & x \in [x_i, x_{i+1}] \end{cases}$$

$$w_1(x) = \frac{x_2 - x}{x_2 - x_1}, x \in [x_1, x_2]$$

$$w_n(x) = \frac{x - x_{n-1}}{x_n - x_{n-1}}, x \in [x_{n-1}, x_n]$$

 $\hat{f}(x) = \sum_{i=1}^n w_i(x) f_i = \frac{x_{i+1} - x}{x_{i+1} - x_i} f_i + \frac{x - x_i}{x_{i+1} - x_i} f_{i+1}$

Stückweise lineare Interpolation

Variante (b)

Auf jedem Intervall $D_i = [x_i, x_{i+1}]$ $i = 1, \dots, n - 1$ werden 2 Funktionen

$$F_i^{(l)}(x) = \frac{x_{i+1} - x}{x_{i+1} - x_i}$$

$$F_i^{(r)}(x) = \frac{x - x_i}{x_{i+1} - x_i}$$

Definiert.

$$\Rightarrow \hat{f}(x) = \sum_{i=1}^{n-1} F_i^{(l)}(x)f_i + F_i^{(r)}(x)f_{i+1}$$

Die Terme $\frac{x_{i+1}-x}{x_{i+1}-x_i}$ und $\frac{x-x_i}{x_{i+1}-x_i}$ stellen die s. g. *baryzentrischen Koordinaten* in \mathbb{R}^1 von $x \in [x_i, x_{i+1}]$ dar.

Stückweise lineare Interpolation

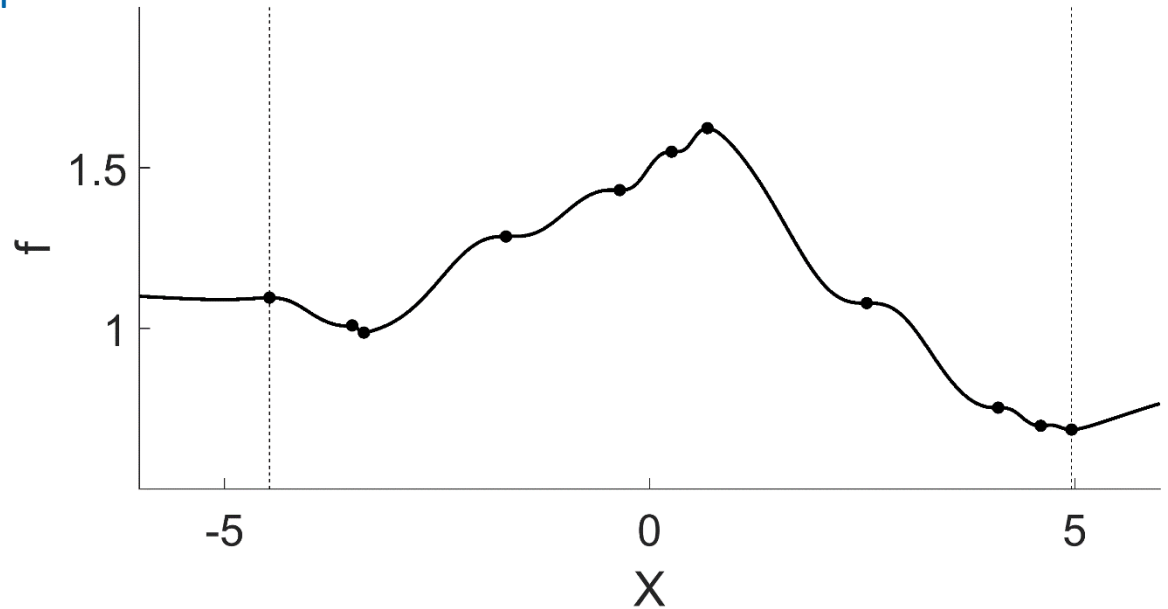
Beide Varianten lassen sich leicht auf \mathbb{R}^n verallgemeinern:

- $n = 2$: lineare Interpolation auf Dreiecken
- $n = 3$: lineare Interpolation auf Tetraedern

Inverse distance weigthing - IDW

Modellannahme:

Ein vorhergesagter Wert für eine Position x wird von allen bekannten $f_i(x_i)$ beeinflusst. Der Einfluss nimmt proportional zum Abstand $d(x, x_i)$ ab.



Inverse distance weigthing - IDW

$$\hat{f}(x) = \frac{\sum_{i=1}^n w_i(x) f_i}{\sum_{i=1}^n w_i(x)}$$

mit

$$w_i(x) = \frac{1}{d(x, x_i)^p}$$

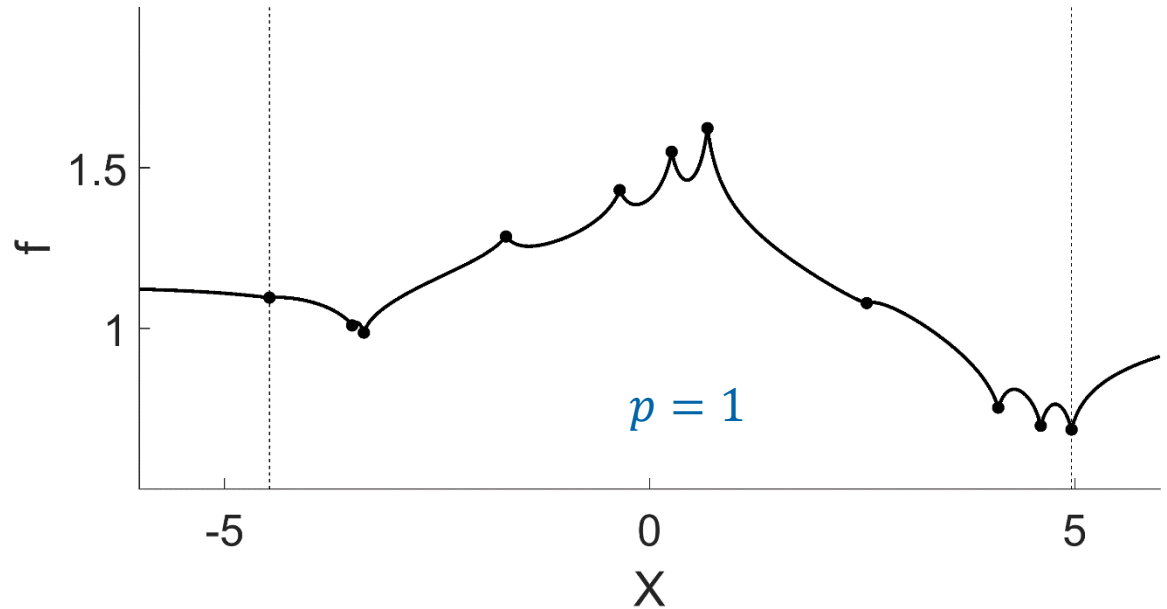
und Lokalisierungs-Parameter p .

Inverse distance weighting - IDW

$$\hat{f}(x) = \frac{\sum_{i=1}^n w_i(x) f_i}{\sum_{i=1}^n w_i(x)}$$

mit

und Lokalisierungs-Param

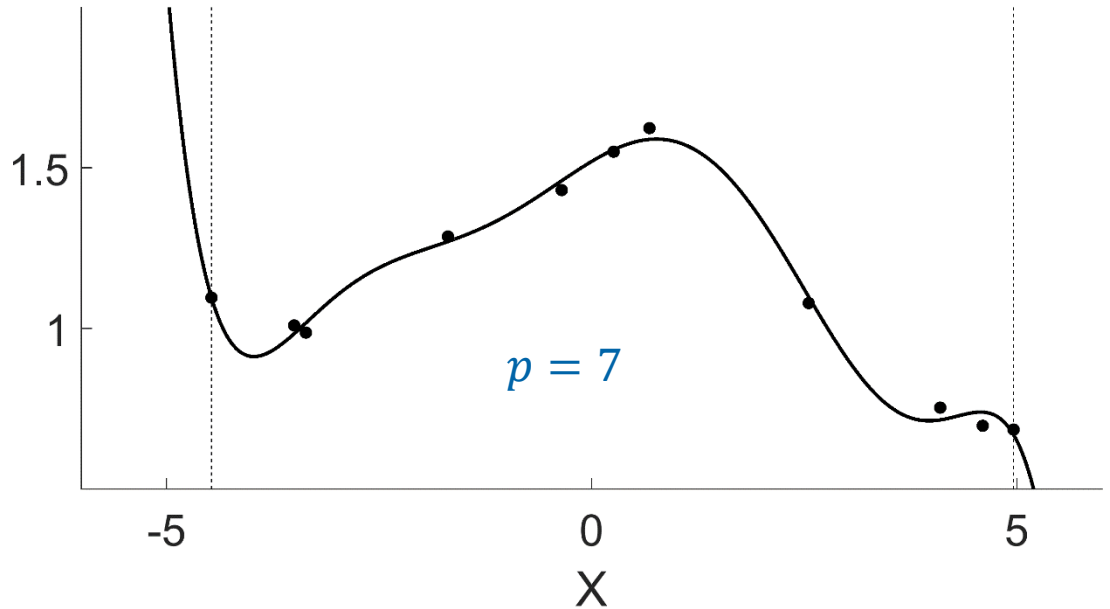


Polynominterpolation/-approximation

Modellannahme:

Ein vorhergesagter Wert für eine Position x ergibt sich aus einem Polynom P vom Grad p der Form:

$$\hat{f}(x) = P(x, p) = \sum_{i=0}^p a_i x^i$$



Lagrange Interpolationspolynom

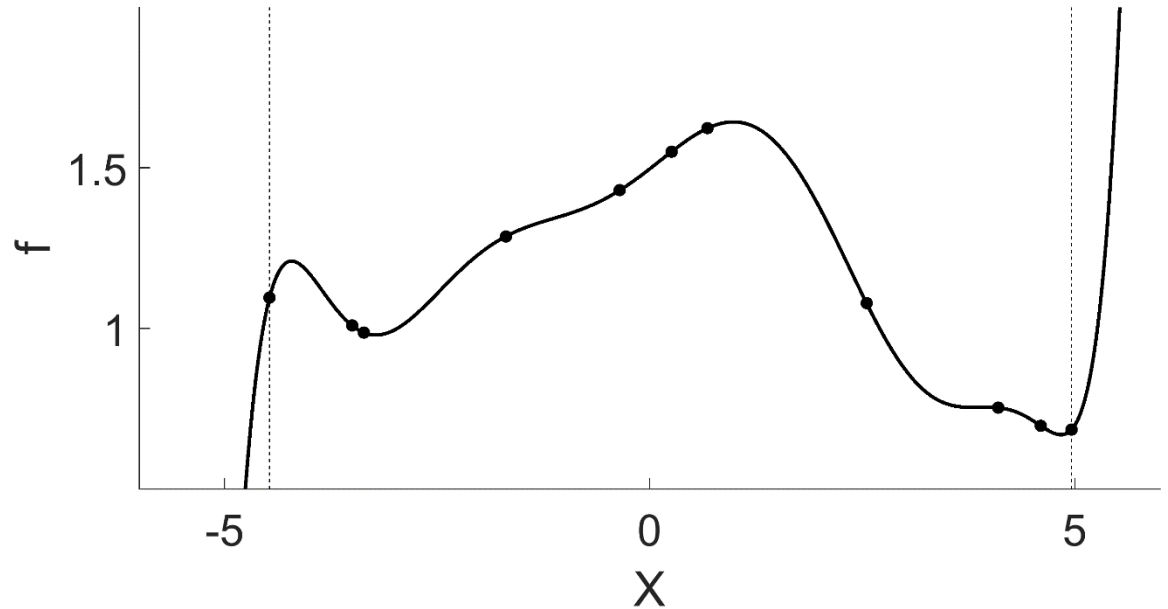
Für n Punkte $(x_i, f_i) \in \mathbb{R}^2$ existiert **genau** ein Polynom P vom Grad $n - 1$ so, dass $P(x_i, n - 1) = f_i$ gilt.

- Polynomgrad hängt direkt von der Anzahl der Daten ab
 - Problem: für hohe Grade neigen Polynome zum starken Oszillieren
 - Für $n > 6$ im Allgemeinen unbrauchbar

Lagrange Interpolationspolynom

Für n Punkte $(x_i, f_i) \in \mathbb{R}^2$ existiert **genau** ein Polynom P vom Grad $n - 1$ so, dass $P(x_i, n - 1) = f_i$ gilt.

- Polynomgrad hängt di
 - Problem: für hohe
 - Für $n > 6$ im Allg



Lagrange Interpolationspolynom

Für n Punkte $(x_i, f_i) \in \mathbb{R}^2$ existiert **genau** ein Polynom P vom Grad $n - 1$ so, dass $P(x_i, n - 1) = f_i$ wie folgt gilt:

$$P(x, n - 1) = L(f; x) = \sum_{i=1}^n f_i L_i^{(n-1)}(x)$$

$$L_i^{(n-1)}(x) = \frac{\prod_{j \neq i} (x - x_j)}{\prod_{j \neq i} (x_i - x_j)} = \prod_{j \neq i} \frac{(x - x_j)}{(x_i - x_j)}$$

mit

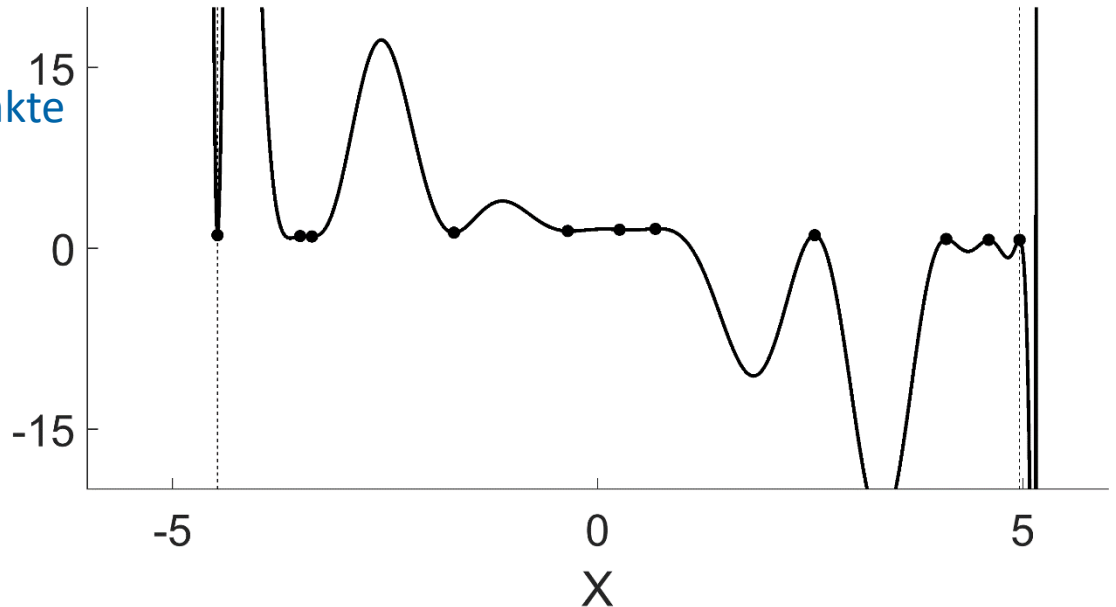
$$\sum_{i=1}^n L_i^{(n-1)}(x) = 1$$

$$L_i^{(n-1)}(x_j) = \begin{cases} 0, & i \neq j \\ 1, & i = j \end{cases}$$

Hermite Interpolationspolynom

Für n Punkte $(x_i, f_i) \in \mathbb{R}^2$ existiert **genau** ein Polynom P vom Grad $2n - 1$ so, dass $P(x_i, 2n - 1) = f_i$ und $P'(x_i, 2n - 1) = f'_i$ gilt.

- Extrem hoher Polynomgrad schon für wenige Datenpunkte
 - Starke Oszillation
- Anstelle von (x_i, f_i) muss (x_i, f_i, f'_i) bekannt sein oder geschätzt werden.



Hermite Interpolationspolynom

Für n Punkte $(x_i, f_i) \in \mathbb{R}^2$ existiert **genau** ein Polynom P vom Grad $2n - 1$ so, dass $P(x_i, 2n - 1) = f_i$ und $P'(x_i, 2n - 1) = f'_i$ wie folgt gilt:

$$P(x, 2n - 1) = H(f, x) = \sum_{i=1}^n [f(x_i)A_i(x) + f'(x_i)B_i(x)]$$

Mit

$$A_i(x) = \left[1 - (x - x_i) \frac{\omega''_n(x_i)}{\omega'_n(x_i)}\right] L_i^2(x)$$

$$B_i(x) = (x - x_i) L_i^2(x)$$

Und

$$\omega_k = \prod_{j=1}^k (x - x_j), k = 1, \dots, n$$

$$\omega_0(x) = 1$$

Polynomapproximation von Grad p

Für n Punkte $(x_i, f_i) \in \mathbb{R}^2$ existiert ein Polynom P vom Grad $p \ll n$ mit $P(x, p) = \sum_{i=0}^p a_i x^i$, welches

$$\sum_{i=1}^n (f(x_i) - P(x_i, p))^2$$

minimiert!

- Approximation, basiert auf der Idee der linearen Regression
- Für $p = n - 1$:
 $P(x, n - 1) = L(f; x)$

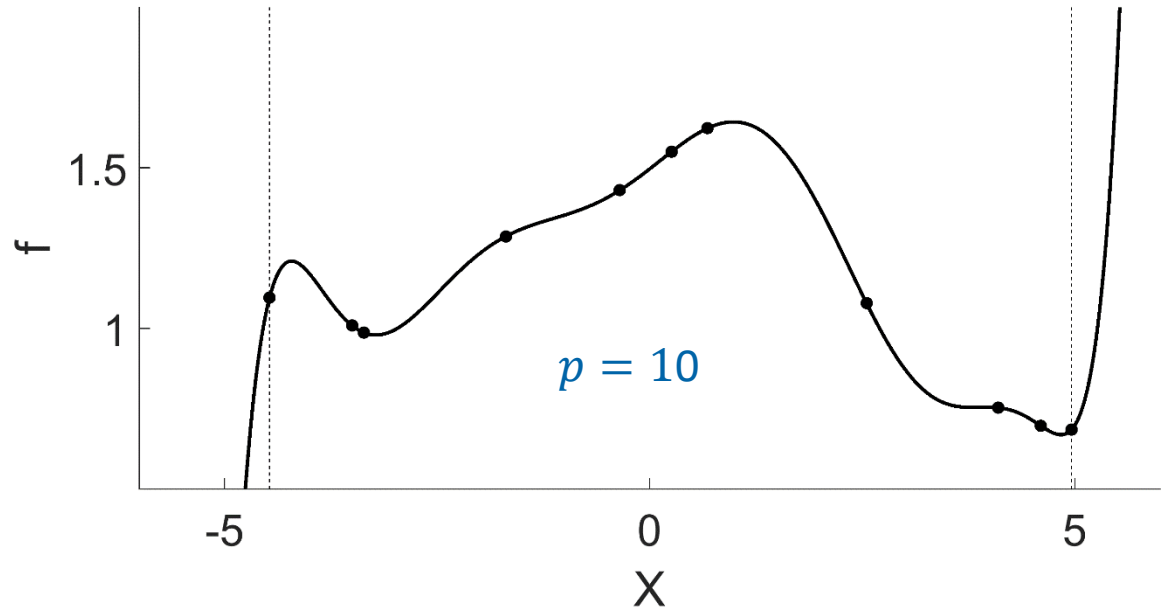
Polynomapproximation von Grad p

Für n Punkte $(x_i, f_i) \in \mathbb{R}^2$ existiert ein Polynom P vom Grad $p \ll n$ mit $P(x, p) = \sum_{i=0}^p a_i x^i$, welches

minimiert!

- Approximation, basiert der Idee der linearen R
- Für $p = n - 1$:

$$P(x, n - 1) = L(f; x)$$



Polynomapproximation von Grad p

Für n Punkte $(x_i, f_i) \in \mathbb{R}^2$ existiert ein Polynom P vom Grad $p \ll n$ mit $P(x, p) = \sum_{i=0}^p a_i x^i$, welches

$$\sum_{i=1}^n (f(x_i) - P(x_i, p))^2$$

minimiert!

Für $n \geq p$ ist das Problem eindeutig lösbar:

$$A\mathbf{x} = \mathbf{b}$$

Wenn man folgende Modellannahme berücksichtigt

$$f(x) = \hat{f}(x) + \varepsilon(x) = a_0 + a_1 x + \varepsilon(x)$$

mit normalverteiltem Term $\varepsilon(x)$ wird aus der einfachen Gradenanpassung an eine Punktwolke die **lineare Regression**. (siehe LV „Datenanalyse/Statistik“ & „Multivariate und Geostatistik“)

Polynomapproximation von Grad p

$$\mathbf{A}\mathbf{x} = \mathbf{b}$$

Mit $n \times p$ Matrix $A_{n \times (p+1)}$

$$A_{n \times (p+1)} = \begin{bmatrix} x_1^p & x_1^{p-1} & \cdots & x_1^0 \\ x_2^p & x_2^{p-1} & \cdots & x_2^0 \\ \vdots & \vdots & \ddots & \vdots \\ x_n^p & x_n^{p-1} & \cdots & x_n^0 \end{bmatrix}$$

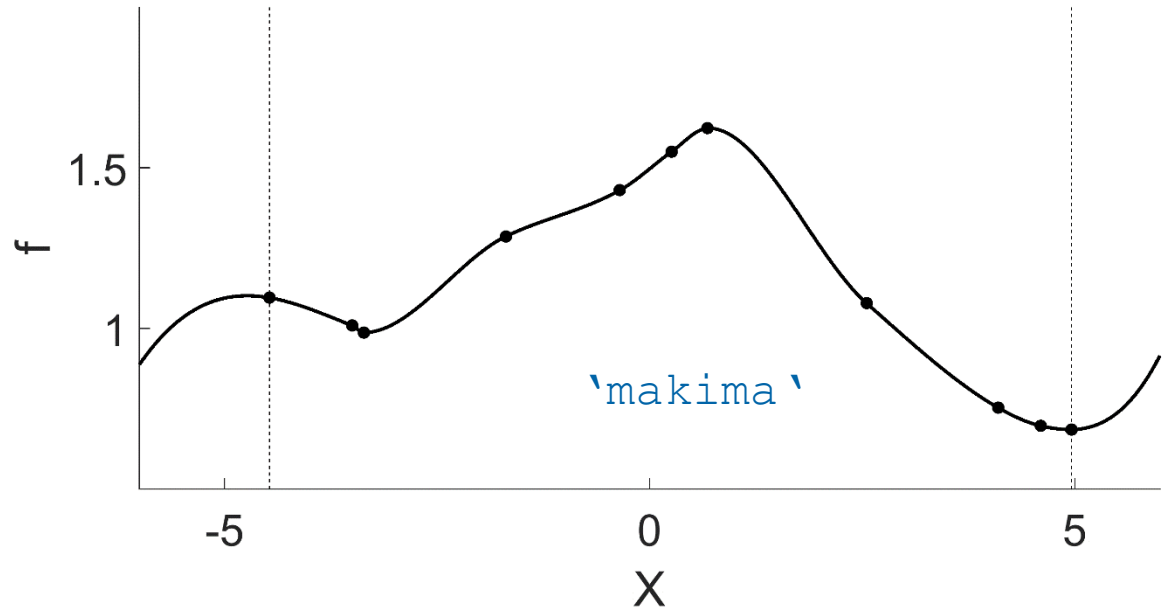
$$\mathbf{x} = \begin{bmatrix} a_p \\ a_{p-1} \\ \vdots \\ a_0 \end{bmatrix}$$

$$\mathbf{b} = \begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ f_n \end{bmatrix}$$

- $\text{rank}(A) = n = (p + 1) \rightarrow \mathbf{x} = \mathbf{A}^{-1}\mathbf{b}$ eindeutig lösbar
- $\text{rank}(A) = (p + 1) < n \rightarrow \mathbf{x} = (\mathbf{A}^t\mathbf{A})^{-1}\mathbf{A}^t\mathbf{b}$ überbestimmt, eindeutig lösbar
- $\text{rank}(A) = n < (p + 1) \rightarrow \mathbf{x} = \mathbf{A}^t(\mathbf{A}\mathbf{A}^t)^{-1}\mathbf{b}$ unterbestimmt, nicht eindeutig lösbar

Polynomiale Splines

$\hat{f}(x)$ kann so konstruiert werden, dass es stückweise, d.h. zwischen zwei Knoten, mit einem Polynom kleinen Grades übereinstimmt, sich global aber nicht durch ein Polynom beschreiben lässt.



Polynomiale Splines

$\hat{f}(x)$ kann so konstruiert werden, dass es stückweise, d.h. zwischen zwei Knoten, mit einem Polynom kleinen Grades übereinstimmt, sich global aber nicht durch ein Polynom beschreiben lässt.

- Dies kann dadurch erreicht werden, dass man die „Glattheitsanforderungen“ in den Knoten verglichen mit einem Polynom zurücknimmt.

Ein *Spline* zu den Knoten v_0, \dots, v_N vom Grad p ist eine Funktion,

- die zwischen zwei Knoten mit einem Polynom vom Grad p übereinstimmt und
 - deren Ableitungen bis einschließlich Ordnung $(p - 1)$ stetig sind.
- Ableitungen höherer Ordnung dürfen in den Knoten unstetig sein!

Polynomiale Splines

Stückweise konstante Interpolation

- Spline vom Grad 0 zu Knoten v_i

Stückweise lineare Interpolation

- Spline vom Grad 1; hier entsprechen die Knoten den Datenpunkten x_i

Kubischer Spline

- Spline vom Grad 3; entspricht einem Polynom 3. Grades (kubisch) zwischen den Knoten und ist an den Knoten mindestens zweimal stetig differenzierbar.

Polynomiale Splines

Zerlegung $\Delta: a = x_0 < x_1 < \dots < x_N = b$ zu den Knoten x_0, x_1, \dots, x_N .

Eine spline Funktion $s(x)$ vom Grad $n \geq 1$ mit Knoten x_0, x_1, \dots, x_N ist eine Funktion auf dem Bereich $(-\infty, \infty)$ die so definiert ist, dass

- $s(x)$ für jedes Intervall (x_i, x_{i+1}) , $0 \leq i \leq N - 1$ und für $(-\infty, a)$ und (b, ∞) ein Polynom von Grad n ist und
- $s(x)$ und seine Ableitungen bis Ordnung $n - 1$ stetig sind.

➤ Die Klasse aller spline Funktionen von Grad n mit Knoten Δ bezeichnet man mit $Sp(n, \Delta)$.

Polynomiale Splines

Eigenschaften

$$s(x) \in Sp(n, \Delta) \Leftrightarrow s'(x) \in Sp(n-1, \Delta)$$

$Sp(n, \Delta) = \{f \mid f^{(n-1)}$ stetig auf $(-\infty, \infty)$, $f^{(n)}$ Treppenfunktion mit Sprüngen an $\Delta\}$

Wenn $f^{(n)}$ stetig und $f \in Sp(n, \Delta)$, dann ist f ein Polynom vom Grad n .

Polynomiale Splines

$s(x) \in Sp(n, \Delta)$ ist charakterisiert durch $N + 2$ Polynome vom Grad n , d.h. mit $n + 1$ zu bestimmenden Koeffizienten und n Glattheitsbedingungen an den $N + 1$ Knoten.

- $(n + 1)(N + 2) - n(N + 1) = n + N + 2$ freier Parameter zur Lösung der Interpolationsaufgabe
- Lineares Interpolationsproblem: suche $s(x) \in Sp(n, \Delta)$ so, dass

$$s(x_i) = f_i, 0 \leq i \leq N$$

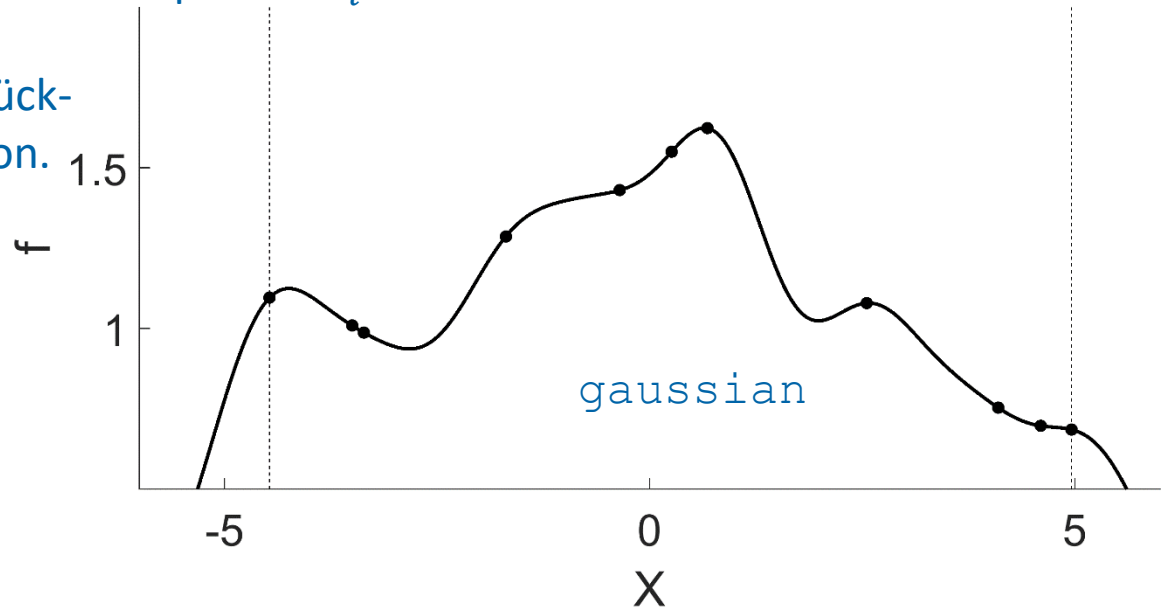
, also $N + 1$ Interpolationsbedingungen erfüllt sind.

Radiale Basisfunktionen

Modellannahme:

$\hat{f}(x)$ kann so konstruiert werden, dass es als Linearkombination glatter, lokal kompakter Funktionen um die Datenpunkte x_i darstellbar ist.

Analog zu Version (a) der stückweisen linearen Interpolation. Anstelle der „hütchenförmigen“ Funktionen w_i werden glatte, glockenförmige Funktionen $\chi(x)$ verwendet.



$$\hat{f}(x) = \sum_{k=1}^n \alpha_k \chi_k(x)$$

Radiale Basisfunktionen

Mögliche $\chi_i(r)$

- Surface $\chi(r) = r^{2n-1}, n \in \mathbb{N}$
- Thin plate $\chi(r) = r^{2n} \log(r), n \in \mathbb{N}$
- Multiquadric $\chi(r) = (r^2 + c^2)^{1/2}, c \in \mathbb{R}$
- Inverse multiquadric $\chi(r) = (r^2 + c^2)^{-1/2}, c \in \mathbb{R}$
- Shifted surface $\chi(r) = (r^2 + c^2)^{n-1/2}, n \in \mathbb{N}, c \in \mathbb{R}$
- Shifted thin plate $\chi(r) = (r^2 + c^2)^n \log(r^2 + c^2), n \in \mathbb{N}, c \in \mathbb{R}$
- Shifted logarithm $\chi(r) = \log(r^2 + c^2), c \in \mathbb{R}$
- **Gaussian** $\chi(r) = e^{(-cr^2)}, c \in \mathbb{R}_+$

$$\chi_k(x) = \chi_k(r_k) \text{ mit } r_k = ||x - x_k||$$

Radiale Basisfunktionen

Für eine Punktmenge $\{x_i, f_i\}$, $1 \leq i \leq N$ und N_b Basispunkten $\{x'_k\}$, $1 \leq k \leq N_b$ lässt sich das Gewicht α_k für jede Basisfunktion χ_k wie folgt bestimmen

$$A = \begin{bmatrix} \chi_1(x_1 - x'_1) & \cdots & \chi_{N_b}(x_1 - x'_{N_b}) \\ \vdots & \chi_k(x_i - x'_k) & \vdots \\ \chi_1(x_N - x'_1) & \cdots & \chi_{N_b}(x_N - x'_{N_b}) \end{bmatrix}$$

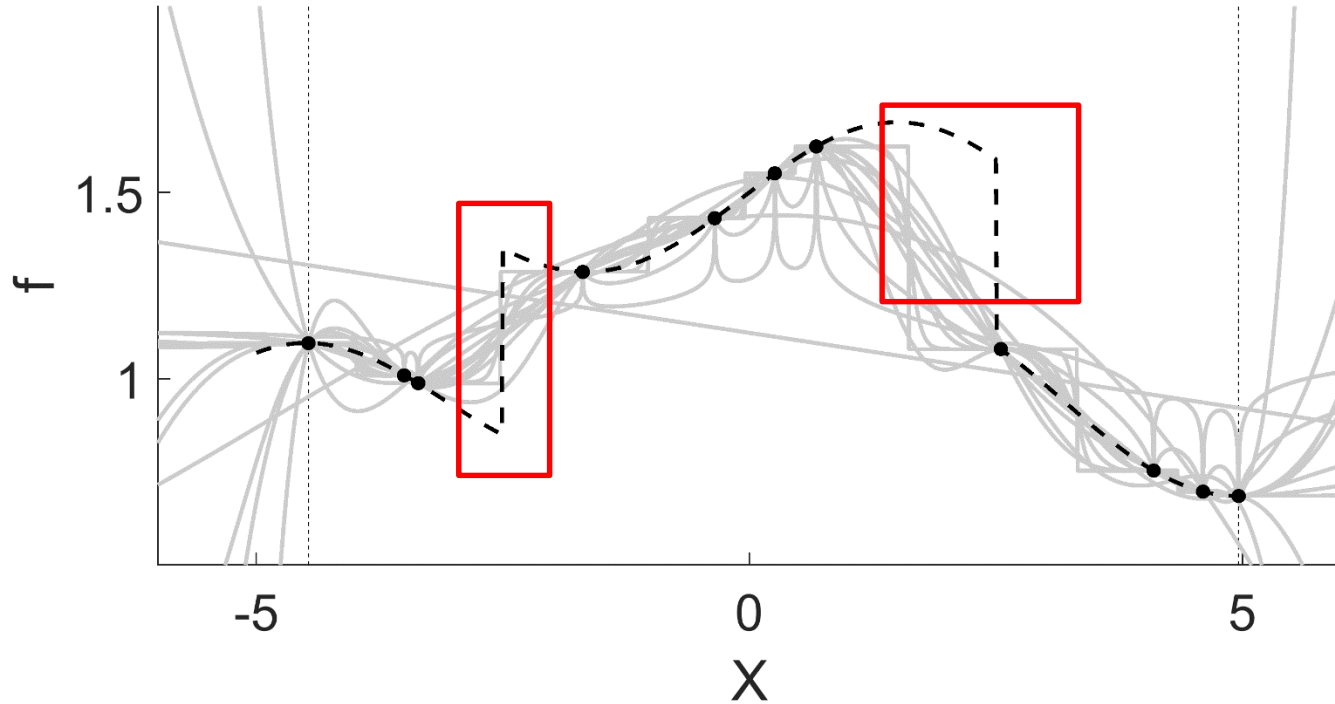
$$\alpha = A^{-1}f \quad (N = N_b)$$

$$\alpha = (A^t A)^{-1} A^t f \quad (N > N_b)$$

$$\Rightarrow \hat{f}(x) = \sum_{k=1}^{N_b} \alpha_k \chi_k(x)$$



Welches Verfahren hat denn nun recht ... ?

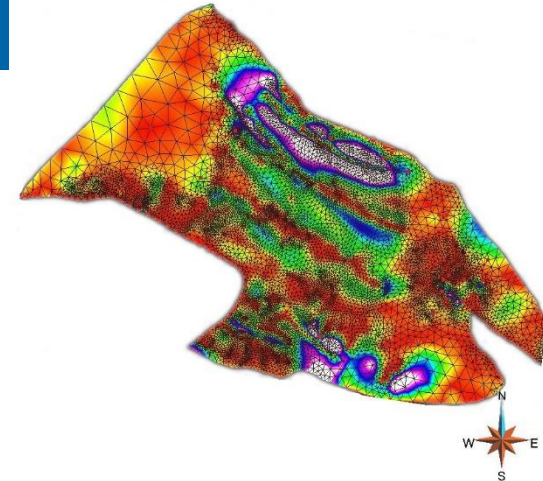


Ergänzungen

- Jedes Verfahren sagt $\bar{f}(x)$ ausgehend von den gegebenen $f_i(x_i)$ unter einer bestimmten Modellannahme voraus: $\hat{f}(x) \rightarrow \bar{f}(x)$
- Viele Verfahren lassen sich zusätzlich zu $f_i(x_i)$ parametrisieren, um das Modell anzupassen.
- Häufig verwendet in der Praxis:
 - Stückweise lineare Interpolation
 - Polynomapproximation
 - Polynomiale Splines
- Alle gezeigten Verfahren lassen sich von 1D auf beliebige Dimensionen erweitern. Für Verfahren, welche im 1D auf Intervallen definiert sind, benötigt man Diskretisierungen des Raumes für höhere Dimensionen.
 - **Tessellations (Vermaschungen)**

Take-home questions:

- 1) Welche Modellannahme verwendet die „stückweise-konstante“ Interpolation? Gehört sie zu den lokalen oder globalen Interpolationsverfahren?
- 2) Wie werden die Interpolationsgewichte bei der IDW-Interpolation bestimmt? Warum gilt dieses Verfahren als „einfachstes“ Interpolationsverfahren?
- 3) Warum sind globale Interpolationspolynome für Probleme mit einer hohen Anzahl von Eingabedaten nicht sinnvoll.
- 4) Sind manche Interpolationsverfahren „korrekter“ als andere?
- 5) Entsprechen die Vorhersagen aus einem Interpolationsverfahren der Realität?



Institut für Geophysik und Geoinformatik
Dr. Peter Menzel
Gustav-Zeuner-Str. 12
09599 Freiberg
Tel. +49(0)3731 39-3815