Nichtparametrische Regression via Neuronaler Netzwerke mithilfe der ReLU-Funktion

Seminar: Mathematische Grundlagen von Statistischem Lernen

Dozentin: Prof. Dr. Rohde

Doktorand: Dario Kieffer

23.06.2020, Lukas Königs

Gliederung

- Nichtparametrische Regression
- Motivation
- Neuronale Netzwerke, Netzwerk-Klasse und Schätzer
- Parallelen und Unterschiede zum Klassifikationsproblem
- Funktionenklasse
- Regressionfunktion
- Theorem 2 mit Beweis
- Theorem 1 mit Beweis-Idee
- Ausblick

nichtparametrische Regression

Regressionsmodell

- n iid Vektoren $X_i \in [0,1]^d$
- n abhängige Variablen $Y_i \in \mathbb{R}$

Definiere für ein $f_0 : [0,1]^d \to \mathbb{R}$

$\forall i \in \mathbb{N} : Y_i = f_0(X_i) + \epsilon_i$ mit ϵ_i standardnormalverteilt und unabhängig von $(X_i)_i$

Regressionsproblem:

Finde in Abhängigkeit von den Daten $((X_1, Y_1), \ldots, (X_n, Y_n))$ einen möglichst guten Schätzer $\hat{f}_n$ für f_0.

lineare Einfachregression

$f_0(x)$ ist eine Gerade: $f_0(x) = \beta_0 + \beta_1 X$

->Struktur ist bis auf zwei unbekannte Parameter festgelegt

->Schätzung der Regressionsfunktion wird auf die Schätzung der unbekannten Parameter zurückgeführt

Problem: Es wird ein linearer Zusammenhang angenommen, ist das korrekt?

nichtparametrische Regression

Es wird keine spezielle funktionale Form der Regressionsfunktion angenommen.

Qualitative Modellannahme: f_0 hinreichend glatt

->funktionale Form der Regressionsfunktion wird aus den Daten bestimmt

(vgl. Fahrmeir L., Kneib T)

Motivation

Deep Learning: visuelle Bilderkennung, Spracherkennung,…

Wesentlich: Lernen mit Daten, einfaches Retraining und hohe Flexibilität

siehe Vorlesung Deep Learning (Prof. Dr. Harms)

Motivation

Approximationstheorie

Neuronale Netzwerke zur Approximierung von Funktionen

-> Dichtheit neuronaler Netzwerkfunktionen in vielen Funktionenräumen

[Cybenko 1989]

-> Multilayer Feedforward Networks are Universal Approximators

[Hornik, K]

Generalisierungsfähigkeit

Achtung!! „Unreasonable Effectiveness of Deep Learning in artificial intelligence“-

Sejnowski, T.

Neuronale Netzwerke

Neuronale Netzwerk-Architektur (L,p)

$L \in \mathbb{Z}_+$ für die Zahl der Hidden Layers

beziehungsweise für die Tiefe

$p = (p_0, \ldots, p_{L+1}) \in \mathbb{N}^{L+2}$ für die Weite

Neuronales Netzwerk (Netzwerkfunktion)

$$f : \mathbb{R}^{p_0} \to \mathbb{R}^{p_{L+1}} \qquad x \mapsto f(x) = W_L \circ \sigma_{V_L} \circ W_{L-1} \circ \sigma_{V_{L-1}} \circ \circ \circ W_1 \circ \sigma_{V_1} \circ W_0 x$$

Netzwerk-Parameter: W_i eine $p_i \times p_{i+1}$ Gewicht-Matrix

$v_i \in \mathbb{R}^{p_i}$ Verschiebungsvektor.

Aktivierungsfunktion

$$\sigma : \mathbb{R} \to \mathbb{R}$$

$$\sigma(x) = \max(x,0)$$

Neuronale Netzwerke

shifted Aktivierungsfunktion

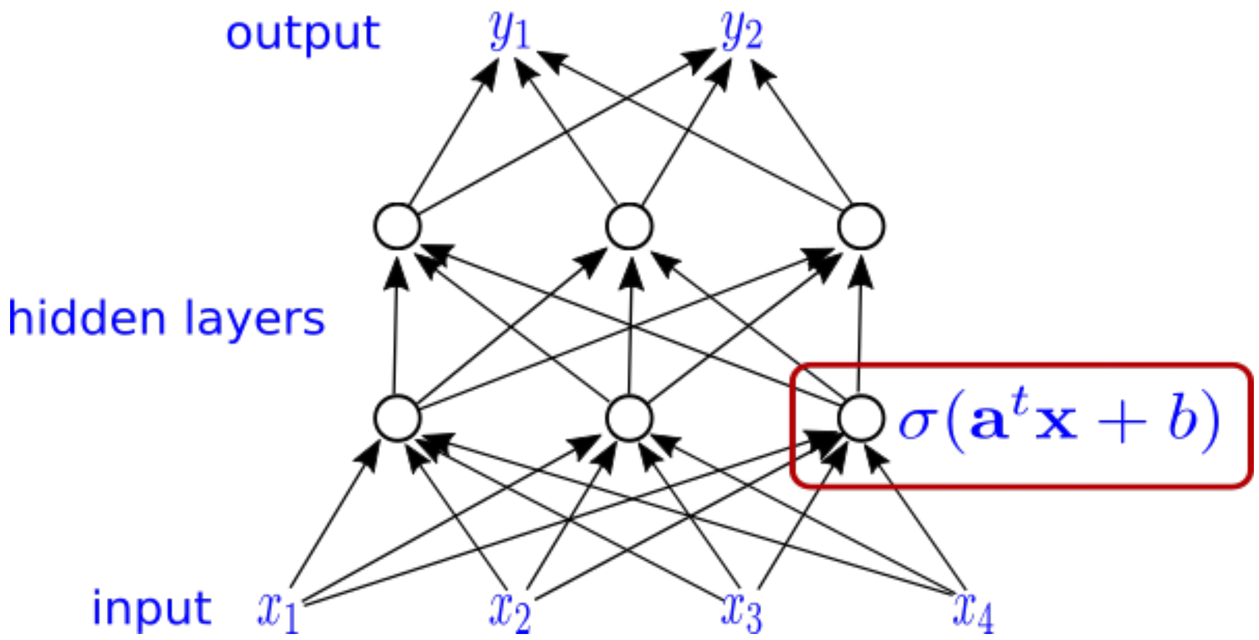
$$\sigma_V : \mathbb{R}^r \to \mathbb{R}^r \text{ für } v = (v_1, \ldots, v_r) \in \mathbb{R}^r$$

$$\sigma_V \begin{pmatrix} y_1 \\ \vdots \\ y_r \end{pmatrix} = \begin{pmatrix} \sigma(y_1 - v_1) \\ \vdots \\ \sigma(y_r - v_r) \end{pmatrix}$$

Parameterwahl

$$p_0 = d \text{ und } p_{L+1} = 1$$

Mathematische Definition ist äquivalent zur grafischen Repräsentation



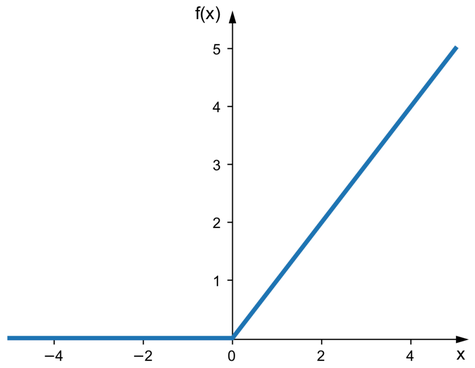
Notation

- x = $(x_1, \ldots, x_d)^\top$ Vektor
- $|x|_p := (\sum_{i=1}^{d} |x_i|^p)^{1/p}$ mit $|x|_\infty := \max_i |x_i|$ und $|x|_0 := \sum_i 1_{(x_i \neq 0)}$
- $\|f\|_p := \|f\|_{L^p(D)}$ sei die L^p-Norm auf D wann immer keine Zweideutigkeit auf dem Definitionsbereich D besteht
- Für zwei Folgen $(a_n)_n$ und $(b_n)_n$:

  $a_n \lesssim b_n$ falls eine Konstante C existiert, sodass: $a_n \leq C b_n$ für alle n

  $a_n \asymp b_n$ falls $a_n \lesssim b_n$ und $b_n \lesssim a_n$
- log_2 für den Logarithmus zur Basis 2
- $\lceil x \rceil$ für die kleinste ganze Zahl größer gleich x

Parameter



- Parameter-Trick:

  Indikatorfunktion leicht durch ReLU-Funktion abschätzbar:

$$x \mapsto \sigma(ax) - \sigma(ax - 1)$$

- in der Praxis: kleine Parameter

  - beschränken die Parameter im Absolutwert durch 1
  - große Tiefe L und große Zahl von potenziellen Netzwerk-Parametern

Netzwerk-Klasse und Schätzer

Netzwerk-Klasse $\mathcal{F}(L, \mathbf{p}) := \{f \text{ Netzwerkfunktion} : \max_{j=0,\dots,L} \|W_j\|_\infty \vee |v_j|_\infty \leq 1\}$

s-besetzte Netzwerk-Klasse $\mathcal{F}(L, \mathbf{p}, s) := \mathcal{F}(L, \mathbf{p}, s, F) := \left\{ f \in \mathcal{F}(L, \mathbf{p}) : \sum_{j=0}^{L} \|W_j\|_0 + |v_j|_0 \leq s, \left\| |f|_\infty \right\|_\infty \leq F \right\}$

Empirische Risikominimierung: $\hat{f}_n \in \arg\min_{f \in \mathcal{F}(L,p,s)} \frac{1}{n} \sum_{i=1}^{n} (Y_i - f(\mathbf{X}_i))^2$

Parallelen und Unterschiede zum Klassifikationsproblem

Erinnerung Klassifikationsproblem:

Seien $(X_1, Y_1), (X_2, Y_2), \ldots \sim P$. Finde in Abhängigkeit von Daten $((X_1, Y_1), \ldots, (X_n, Y_n))$ einen möglichst guten Schätzer von $f_0 := \arg\max_{y \in \mathcal{Y}} P(Y = y \,|\, X = \cdot\,)$.

- Klassifikation: endlich viele Werte; Regression: Y kann unendlich viele Werte annehmen (in $\mathbb{R}$)

- Die Varianz $V[X_i, Y_i]$ (Ungenauigkeit) kann von dem Wert X_i abhängen. Im Regressionsmodell ist $V[Y_i \,|\, X_i] = V[\epsilon_i] = 1$ unabhängig von der Position X_i.

Aber Begriffe lassen sich ähnlich verwenden:

- 'estimation error': $$\Delta_n(\hat{f}_n, f_0) = \Delta_n(\hat{f}_n, f_0, \mathcal{F}(L, p, s, F)) := E_{f_0}\left[\frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{f}_n(\mathbf{X}_i))^2 - \inf_{f \in \mathcal{F}(L, \mathbf{p}, s, F)} \frac{1}{n}\sum_{i=1}^{n}(Y_i - f(\mathbf{X}_i))^2\right]$$

  (vergleiche 3. Vortrag: 'Algorithmus-Risiko')

- Klassifikationsproblem: Minimiere Risiko bezüglich des Bayes-Klassifizierers. Hier: Minimiere Risiko bezüglich f_0, also

$$R(\hat{f}_n, f_0) := E_{f_0}[(\hat{f}(\mathbf{X}) - f_o(\mathbf{X}))^2]$$

Funktionenklasse

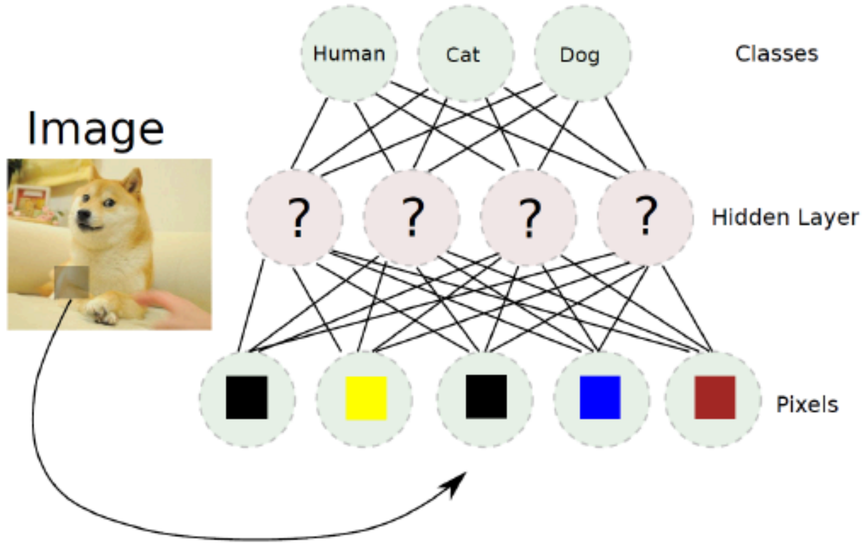
Regressionsfunktion Hölder-stetig mit Konstante β

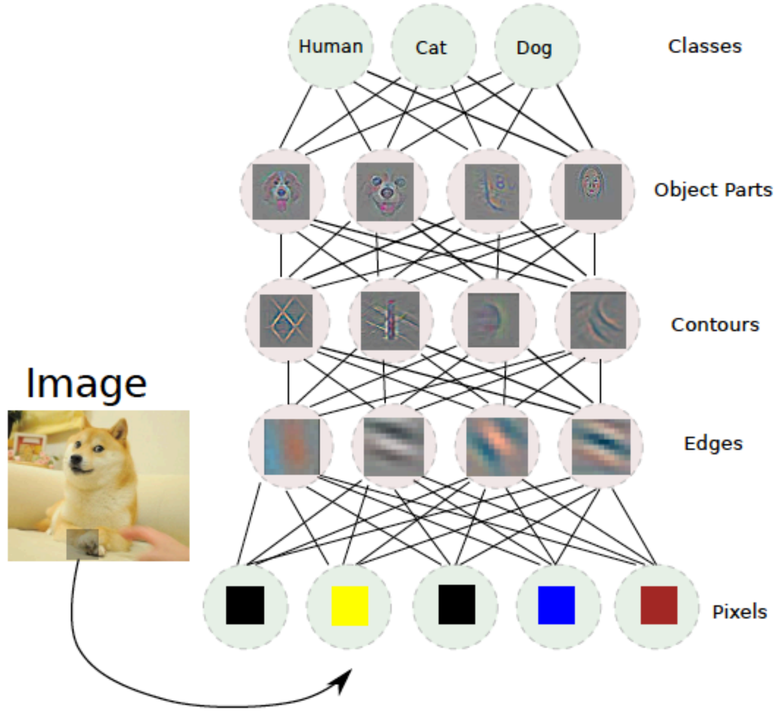
MiniMax-Rate $n^{-2\beta/(2\beta+d)}$ für den Vorhersage-Fehler:

$$\inf_{\hat{f}_n} \sup_{f_0 \in \mathcal{G}(q,d,t,\beta,K)} R(\hat{f}_n, f_0)$$

curse of dimensionality

Hierarchische Struktur





Regressionfunktion

Regressionsfunktion $f_0 = g_q \circ g_{q-1} \circ \ldots \circ g_1 \circ g_0$

- mit $g_i : [a_i, b_i]^{d_i} \to [a_{i+1}, b_{i+1}]^{d_{i+1}}$
- g_{ij} β_i-Hölder-Stetig &
- g_{ij} t_i-variat

Hölder-Ball $\mathscr{C}(D, K) = \left\{ f : D \subset \mathbb{R}^r \to \mathbb{R} : \sum_{\alpha:|\alpha|<\beta} \|\partial^\alpha f\|_\infty + \sum_{\alpha:|\alpha|=\lfloor\beta\rfloor} \sup_{x,y\in D;x\neq y} \frac{|\partial^\alpha f(x) - \partial^\alpha f(y)|}{|x-y|_\infty^{\beta-\lfloor\beta\rfloor}} \leq K \right\}$

=> $g_{ij} \in \mathscr{C}_{t_i}^{\beta_i}([a_i, b_i]^{t_i}, K_i)$

also zugrundeliegender **Funktionenraum**

$$\mathscr{G}(q, \mathbf{d}, \mathbf{t}, \beta, K) := \left\{ f = g_q \circ \ldots \circ g_0 : g_i = (g_{ij})_j : [a_i, b_i]^{d_i} \to [a_{i+1}, b_{i+1}]^{d_{i+1}}, g_{ij} \in \mathscr{C}_{t_i}^{\beta_i}([a_i, b_i]^{t_i}, K), \text{ für ein } |a_i|, |b_i| \leq K \right\}$$

effektive Stetigkeit-Indizes

$$\beta_i^* := \beta_i \prod_{l=i+1}^{q} (\beta_l \wedge 1)$$

Rate

$$\phi_n := \max_{i=0,\dots,q} n^{-\frac{2\beta_i^*}{2\beta_i^* + t_i}}$$

Theorem 2

Theorem 5

Betrachte eine beliebige beliebige Funktion $f \in C_r^\beta([0,1]^r, K)$, ganze Zahlen $m \geq 1$ und$N \geq (\beta+1)^r \vee (K+1)e^r$.

Dann existiert ein Netzwerk $\tilde{f} \in \mathcal{F}(L, (r, 6(r+\lceil\beta\rceil)N, \ldots, 6(r+\lceil\beta\rceil)N, s, \infty)$ mit der Tiefe

$L = 8 + (m+5)(1+\lceil\log_2(r \vee \beta)\rceil)$ und Parameterzahl $s \leq 141(r+\beta+1)^{(3+r)}N(m+6)$, sodass

$$\|\tilde{f} - f\|_{L^\infty([0,1]^r)} \leq (2K+1)(1+r^2+\beta^2)6^r N 2^{-m} + K3^\beta N^{-\frac{\beta}{r}}.$$

Beweis-Idee

siehe Vorlesung Deep Learning von Dr. Harms

Lemma 3

Für die Funktion $f = g_q \circ \ldots \circ g_0 = h_q \circ \ldots \circ h_0$ und die Funktion $\tilde{h}_i$ gibt Lemma 3 folgende Abschätzung:

$$\|h_q \circ \ldots \circ h_0 - \tilde{h}_q \circ \ldots \circ \tilde{h}_0\|_{L^\infty([0,1]^d)} \leq K_q \prod_{l=0}^{q-1} (2K_l)^{\beta_{l+1}} \sum_{i=0}^{q} \| |h_i - \tilde{h}_i|_\infty \|_{L^\infty([0,1]^{d_i})}^{\prod_{l=i+1}^{q} \beta_l \wedge 1}.$$

Theorem 1

Betrachtet wird das d-variate nichtparametrische Regressionsmodell für eine zusammengesetzte Regressionsfunktion in der Klasse $\mathcal{G}(q, \mathbf{d}, \mathbf{t}, \beta, K)$. Sei $\hat{f}_n$ ein Schätzer mit Werten aus der Netzwerkklasse $\mathcal{F}(L, (p_i)_{i=0,\dots,L+1}, s, F)$, welcher den Voraussetzungen

(i) $F \geq \max(K,1)$,

(ii) $\sum_{i=0}^{q} \log_2(4t_i \vee 4\beta_i)\log_2 n \leq L \lesssim n\phi_n$,

(iii) $n\phi_n \lesssim \min_{i=1,\dots,L} p_i$

(iv) $s \asymp n\phi_n \log n$.

Es existieren Konstanten C, C', welche nur von $q, \mathbf{d}, \mathbf{t}, \beta, F$ abhängen, sodass falls

$\Delta_n(\hat{f}_n, f_0) \leq C\phi_n L \log^2 n$, **dann**

(1) $R(\hat{f}_n, f_0) \leq C'\phi_n L \log^2 n$,

Und falls

$\Delta_n(\hat{f}_n, f_0) \geq C\phi_n L \log^2 n$, **dann**

(2) $\frac{1}{C'}\Delta_n(\hat{f}_n, f_0) \leq R(\hat{f}_n, f_0) \leq C'\Delta_n(\hat{f}_n, f_0)$.

Beweis-Idee

$$(1-\epsilon)^2\Delta_n(\hat{f}_n, f_0) - \tau_{\epsilon,n} \leq R(\hat{f}_n, f_0) \leq (1+\epsilon)^2\left(\inf_{f\in\mathcal{F}(L,\mathbf{p},s,F)} \|f-f_0\|_\infty^2 + \Delta_n(\hat{f}_n, f_0)\right) + \tau_{\epsilon,n}$$

-> Zeige $\tau_{\epsilon,n} \leq C'\phi_n L \log 2n$

Dann: $(1-\epsilon)^2\Delta_n(\hat{f}_n, f_0) - C'\phi_n L \log 2n \leq R(\hat{f}_n, f_0) \leq (1+\epsilon)^2\left(\inf_{f\in\mathcal{F}(L,\mathbf{p},s,F)} \|f-f_0\|_\infty^2 + \Delta_n(\hat{f}_n, f_0)\right) + C'\phi_n L \log 2n$

-> Wähle $C = 8C'$, $\epsilon = 1/2$ für die untere Schranke, $\epsilon = 1$ für die obere Schranke

Damit ist die untere Schranke in (2) bewiesen.

Für die oberen Schranken:

-> forme Regressionsfunktion um

-> konstruiere und erweitere die Netzwerk-Klasse mittels Vergrößerung, Komposition von Netzwerken, Hinzufügen von Layern in den Netzwerken, Parallelisieren von Netzwerken

-> bis hin zu der Netzwerk-Klasse $\mathcal{F}(L, p, s)$

-> $\inf_{f^*\in\mathcal{F}(L,p,s)} \|f^*-f_0\|_\infty^2 \leq C' \max_{i=0,\dots,q} N^{-\frac{2\beta_i^*}{t_i}} \leq C' \max_{i=0,\dots,q} c^{-\frac{2\beta_i^*}{t_i}} n^{-\frac{2\beta_i^*}{2\beta_i^*+t_i}}$

-> $\inf_{f^*\in\mathcal{F}(L,p,s)} \|f^*-f_0\|_\infty^2 \leq C\phi_n$

Ausblick

Folgerung 1

Sei $\tilde{f}_n \in \arg\min_{f \in \mathcal{F}(L,\mathbf{p},s,F)} \sum_{i=1}^{n} (Y_i - f(\mathbf{X}_i))^2$ ein empirischer Risiko-Minimierer. Es gelten dieselben Voraussetzungen wie in Theorem 1. Dann existiert eine Konstante C', welche lediglich von q, $\mathbf{d}$, $\mathbf{t}$, β, F abhängt, sodass

$$R(\tilde{f}_n, f_0) \leq C' \phi_n L \log^2 n.$$

Theorem 3

Minimax Schätzrate für den Vorhersagefehler

Vielen Dank für Ihre Aufmerksamkeit!

Quellen

- Cybenko,G. (1989) Approximation by superpositions of asigmoidal function. In: Mathematics of Control, Signals and Systems 2, 4; 303–314.
- Fahrmeir L., Kneib T., Lang S. (2009) Nichtparametrische Regression. In: Regression. Statistik und ihre Anwendungen. Springer, Berlin, Heidelberg. 40-41
- Hornik, K., Stinchcombe, M., and White, H. (1989) Multilayer feedforward networks are universal approximators. In: Neural Networks 2, 5; 359 – 366.
- Hornik, K., Stinchcombe, M., and White, H. (1990) Universal approximation of an unknown mapping and its derivatives using multilayer feedforward networks. In: Neural Networks 3, 5; 551 – 560
- Schmidt-Hieber, J. (2020) Nonparametric Regression using Deep Neural Networks with ReLU Activation function.
- Sejnowski, T. (2020) The unreasonable effectiveness of deep learning in artificial intelligence.
- Zeiler, M., Fergus, R. (2013) Visualizing and Understanding Convolutional Networks.