Über Finanzderivate, den Fluch der Dimensionalität und künstliche Intelligenz

Arnulf Jentzen (Departement für Mathematik, ETH Zürich)

ETH Zürich, Raum HG E7, Rämistrasse 101, 8092 Zürich

14:00-15:00, 18. Januar 2018

Finanzderivat (von Basiswerten abgeleitetes Finanzprodukt)

Eine **europäische Verkaufsoption** ist ein Vertrag zwischen zwei Parteien,

- dem Aussteller der europäischen Verkaufsoption und
- dem Inhaber der europäischen Verkaufsoption,

welche dem Inhaber der Verkaufsoption das Recht, aber nicht die Pflicht, einräumt zu einem vertraglich festgelegten Zeitpunkt $T > 0$ einen vertraglich festgelegten Wert (Basiswert) zu einem vertraglich festgelegten Preis (Ausübungspreis) $K > 0$ zu verkaufen.

Beispiele für Basiswerte: Aktien, Indizes, Anleihen, Rohstoffe, Devisen, Zinsen

Profit durch europäische Verkaufsoption: Sei $x_T \geq 0$ Preis des Basiswertes zur Zeit T.

- Der Fall $x_T > K$ liefert Profit $K - x_T$.
- Der Fall $x_T \leq K$ liefert Profit 0.

Insgesamt also Profit $\max\{K - x_T, 0\}$ (Preis der Verkaufsoption zur Zeit T).

Derivatebewertung: Preis der Verkaufsoption zur Zeit 0.

Finanzderivat (von Basiswerten abgeleitetes Finanzprodukt)

Eine **europäische Verkaufsoption** ist ein Vertrag zwischen zwei Parteien,

- dem Aussteller der europäischen Verkaufsoption und
- dem Inhaber der europäischen Verkaufsoption,

welche dem Inhaber der Verkaufsoption das Recht, aber nicht die Pflicht, einräumt zu einem vertraglich festgelegten Zeitpunkt $T > 0$ einen vertraglich festgelegten Wert (Basiswert) zu einem vertraglich festgelegten Preis (Ausübungspreis) $K > 0$ zu verkaufen.

Beispiele für Basiswerte: Aktien, Indizes, Anleihen, Rohstoffe, Devisen, Zinsen

Profit durch europäische Verkaufsoption: Sei $x_T \geq 0$ Preis des Basiswertes zur Zeit T.

- Der Fall $x_T > K$ liefert Profit $K - x_T$.
- Der Fall $x_T \leq K$ liefert Profit 0.

Insgesamt also Profit $\max\{K - x_T, 0\}$ (Preis der Verkaufsoption zur Zeit T).

Derivatebewertung: Preis der Verkaufsoption zur Zeit 0.

Finanzderivat (von Basiswerten abgeleitetes Finanzprodukt)

Eine **europäische Verkaufsoption** ist ein Vertrag zwischen zwei Parteien,

- dem Aussteller der europäischen Verkaufsoption und
- dem Inhaber der europäischen Verkaufsoption,

welche dem Inhaber der Verkaufsoption das Recht, aber nicht die Pflicht, einräumt zu einem vertraglich festgelegten Zeitpunkt $T > 0$ einen vertraglich festgelegten Wert (Basiswert) zu einem vertraglich festgelegten Preis (Ausübungspreis) $K > 0$ zu verkaufen.

Beispiele für Basiswerte: Aktien, Indizes, Anleihen, Rohstoffe, Devisen, Zinsen

Profit durch europäische Verkaufsoption: Sei $x_T \geq 0$ Preis des Basiswertes zur Zeit T.

- Der Fall $x_T > K$ liefert Profit $K - x_T$.
- Der Fall $x_T \leq K$ liefert Profit 0.

Insgesamt also Profit $\max\{K - x_T, 0\}$ (Preis der Verkaufsoption zur Zeit T).

Derivatebewertung: Preis der Verkaufsoption zur Zeit 0.

Finanzderivat (von Basiswerten abgeleitetes Finanzprodukt)

Eine **europäische Verkaufsoption** ist ein Vertrag zwischen zwei Parteien,

- dem Aussteller der europäischen Verkaufsoption und
- dem Inhaber der europäischen Verkaufsoption,

welche dem Inhaber der Verkaufsoption das Recht, aber nicht die Pflicht, einräumt zu einem vertraglich festgelegten Zeitpunkt $T > 0$ einen vertraglich festgelegten Wert (Basiswert) zu einem vertraglich festgelegten Preis (Ausübungspreis) $K > 0$ zu verkaufen.

Beispiele für Basiswerte: Aktien, Indizes, Anleihen, Rohstoffe, Devisen, Zinsen

Profit durch europäische Verkaufsoption: Sei $x_T \geq 0$ Preis des Basiswertes zur Zeit T.

- Der Fall $x_T > K$ liefert Profit $K - x_T$.
- Der Fall $x_T \leq K$ liefert Profit 0.

Insgesamt also Profit $\max\{K - x_T, 0\}$ (Preis der Verkaufsoption zur Zeit T).

Derivatebewertung: Preis der Verkaufsoption zur Zeit 0.

Finanzderivat (von Basiswerten abgeleitetes Finanzprodukt)

Eine **europäische Verkaufsoption** ist ein Vertrag zwischen zwei Parteien,

- dem Aussteller der europäischen Verkaufsoption und
- dem Inhaber der europäischen Verkaufsoption,

welche dem Inhaber der Verkaufsoption das Recht, aber nicht die Pflicht, einräumt zu einem vertraglich festgelegten Zeitpunkt $T > 0$ einen vertraglich festgelegten Wert (Basiswert) zu einem vertraglich festgelegten Preis (Ausübungspreis) $K > 0$ zu verkaufen.

Beispiele für Basiswerte: Aktien, Indizes, Anleihen, Rohstoffe, Devisen, Zinsen

Profit durch europäische Verkaufsoption: Sei $x_T \geq 0$ Preis des Basiswertes zur Zeit T.

- Der Fall $x_T > K$ liefert Profit $K - x_T$.
- Der Fall $x_T \leq K$ liefert Profit 0.

Insgesamt also Profit $\max\{K - x_T, 0\}$ (Preis der Verkaufsoption zur Zeit T).

Derivatebewertung: Preis der Verkaufsoption zur Zeit 0.

Finanzderivat (von Basiswerten abgeleitetes Finanzprodukt)

Eine **europäische Verkaufsoption** ist ein Vertrag zwischen zwei Parteien,

- dem Aussteller der europäischen Verkaufsoption und
- dem Inhaber der europäischen Verkaufsoption,

welche dem Inhaber der Verkaufsoption das Recht, aber nicht die Pflicht, einräumt zu einem vertraglich festgelegten Zeitpunkt $T > 0$ einen vertraglich festgelegten Wert (Basiswert) zu einem vertraglich festgelegten Preis (Ausübungspreis) $K > 0$ zu verkaufen.

Beispiele für Basiswerte: Aktien, Indizes, Anleihen, Rohstoffe, Devisen, Zinsen

Profit durch europäische Verkaufsoption: Sei $x_T \geq 0$ Preis des Basiswertes zur Zeit T.

- Der Fall $x_T > K$ liefert Profit $K - x_T$.
- Der Fall $x_T \leq K$ liefert Profit 0.

Insgesamt also Profit $\max\{K - x_T, 0\}$ (Preis der Verkaufsoption zur Zeit T).

Derivatebewertung: Preis der Verkaufsoption zur Zeit 0.

Finanzderivat (von Basiswerten abgeleitetes Finanzprodukt)

Eine **europäische Verkaufsoption** ist ein Vertrag zwischen zwei Parteien,

- dem Aussteller der europäischen Verkaufsoption und
- dem Inhaber der europäischen Verkaufsoption,

welche dem Inhaber der Verkaufsoption das Recht, aber nicht die Pflicht, einräumt zu einem vertraglich festgelegten Zeitpunkt $T > 0$ einen vertraglich festgelegten Wert (Basiswert) zu einem vertraglich festgelegten Preis (Ausübungspreis) $K > 0$ zu verkaufen.

Beispiele für Basiswerte: Aktien, Indizes, Anleihen, Rohstoffe, Devisen, Zinsen

Profit durch europäische Verkaufsoption: Sei $x_T \geq 0$ Preis des Basiswertes zur Zeit T.

- Der Fall $x_T > K$ liefert Profit $K - x_T$.
- Der Fall $x_T \leq K$ liefert Profit 0.

Insgesamt also Profit $\max\{K - x_T, 0\}$ (Preis der Verkaufsoption zur Zeit T).

Derivatebewertung: Preis der Verkaufsoption zur Zeit 0.

Finanzderivat (von Basiswerten abgeleitetes Finanzprodukt)

Eine **europäische Verkaufsoption** ist ein Vertrag zwischen zwei Parteien,

- dem Aussteller der europäischen Verkaufsoption und
- dem Inhaber der europäischen Verkaufsoption,

welche dem Inhaber der Verkaufsoption das Recht, aber nicht die Pflicht, einräumt zu einem vertraglich festgelegten Zeitpunkt $T > 0$ einen vertraglich festgelegten Wert (Basiswert) zu einem vertraglich festgelegten Preis (Ausübungspreis) $K > 0$ zu verkaufen.

Beispiele für Basiswerte: Aktien, Indizes, Anleihen, Rohstoffe, Devisen, Zinsen

Profit durch europäische Verkaufsoption: Sei $x_T \geq 0$ Preis des Basiswertes zur Zeit T.

- Der Fall $x_T > K$ liefert Profit $K - x_T$.
- Der Fall $x_T \leq K$ liefert Profit 0.

Insgesamt also Profit $\max\{K - x_T, 0\}$ (Preis der Verkaufsoption zur Zeit T).

Derivatebewertung: Preis der Verkaufsoption zur Zeit 0.

Finanzderivat (von Basiswerten abgeleitetes Finanzprodukt)

Eine **europäische Verkaufsoption** ist ein Vertrag zwischen zwei Parteien,

- dem Aussteller der europäischen Verkaufsoption und
- dem Inhaber der europäischen Verkaufsoption,

welche dem Inhaber der Verkaufsoption das Recht, aber nicht die Pflicht, einräumt zu einem vertraglich festgelegten Zeitpunkt $T > 0$ einen vertraglich festgelegten Wert (Basiswert) zu einem vertraglich festgelegten Preis (Ausübungspreis) $K > 0$ zu verkaufen.

Beispiele für Basiswerte: Aktien, Indizes, Anleihen, Rohstoffe, Devisen, Zinsen

Profit durch europäische Verkaufsoption: Sei $x_T \geq 0$ Preis des Basiswertes zur Zeit T.

- Der Fall $x_T > K$ liefert Profit $K - x_T$.
- Der Fall $x_T \leq K$ liefert Profit 0.

Insgesamt also Profit $\max\{K - x_T, 0\}$ (Preis der Verkaufsoption zur Zeit T).

Derivatebewertung: Preis der Verkaufsoption zur Zeit 0.

Geeignete Modellannahmen liefern:

Funktionaler Zusammenhang zwischen Preise der Basiswerte und Preis des Derivates. **Wertfunktion**

$$u\colon [0,T] \times \mathbb{R}^d \to \mathbb{R}$$

für Derivat mit $d \in \mathbb{N} = \{1, 2, 3, \dots\}$ Basiswerten. Preis des Derivats

$$u(t,x) \in \mathbb{R}$$

zur Zeit $t \in [0,T]$ in Abhängigkeit der Preise $x = (x_1, \dots, x_d) \in \mathbb{R}^d$ der Basiswerte. Wertfunktion erfüllt **partielle Differentialgleichung**

$$\tfrac{\partial u}{\partial t}(t,x) + f\big(x, u(t,x), (\nabla_x u)(t,x), (\operatorname{Hess}_x u)(t,x)\big) = 0 \qquad \text{(PDGL)}$$

mit $u(T,x) = g(x)$ für $t \in [0,T)$, $x \in \mathbb{R}^d$ mit geeigneten Funktionen $f\colon \mathbb{R}^d \times \mathbb{R} \times \mathbb{R}^d \times \mathbb{R}^{d\times d} \to \mathbb{R}$ und $g\colon \mathbb{R}^d \to \mathbb{R}$. **Ziel:** Näherungsweise lösen von (PDGL). Beispiel: **Wärmeleitungsgleichung**

$$\tfrac{\partial u}{\partial t}(t,x) + (\Delta_x u)(t,x) = 0.$$

Klassische Näherungsverfahren für (PDGL) leiden unter **Fluch der Dimensionalität**.

Geeignete Modellannahmen liefern:

Funktionaler Zusammenhang zwischen Preise der Basiswerte und Preis des Derivates. **Wertfunktion**

$$u\colon [0,T] \times \mathbb{R}^d \to \mathbb{R}$$

für Derivat mit $d \in \mathbb{N} = \{1, 2, 3, \dots\}$ Basiswerten. Preis des Derivats

$$u(t,x) \in \mathbb{R}$$

zur Zeit $t \in [0,T]$ in Abhängigkeit der Preise $x = (x_1, \dots, x_d) \in \mathbb{R}^d$ der Basiswerte. Wertfunktion erfüllt **partielle Differentialgleichung**

$$\frac{\partial u}{\partial t}(t,x) + f\big(x, u(t,x), (\nabla_x u)(t,x), (\operatorname{Hess}_x u)(t,x)\big) = 0 \qquad \text{(PDGL)}$$

mit $u(T,x) = g(x)$ für $t \in [0,T)$, $x \in \mathbb{R}^d$ mit geeigneten Funktionen $f\colon \mathbb{R}^d \times \mathbb{R} \times \mathbb{R}^d \times \mathbb{R}^{d\times d} \to \mathbb{R}$ und $g\colon \mathbb{R}^d \to \mathbb{R}$. **Ziel:** Näherungsweise lösen von (PDGL). Beispiel: **Wärmeleitungsgleichung**

$$\frac{\partial u}{\partial t}(t,x) + (\Delta_x u)(t,x) = 0.$$

Klassische Näherungsverfahren für (PDGL) leiden unter **Fluch der Dimensionalität**.

Geeignete Modellannahmen liefern:

Funktionaler Zusammenhang zwischen Preise der Basiswerte und Preis des Derivates. **Wertfunktion**

$$u \colon [0, T] \times \mathbb{R}^d \to \mathbb{R}$$

für Derivat mit $d \in \mathbb{N} = \{1, 2, 3, \dots\}$ Basiswerten. Preis des Derivats

$$u(t, x) \in \mathbb{R}$$

zur Zeit $t \in [0, T]$ in Abhängigkeit der Preise $x = (x_1, \dots, x_d) \in \mathbb{R}^d$ der Basiswerte. Wertfunktion erfüllt **partielle Differentialgleichung**

$$\frac{\partial u}{\partial t}(t, x) + f\big(x, u(t, x), (\nabla_x u)(t, x), (\mathrm{Hess}_x\, u)(t, x)\big) = 0 \qquad \text{(PDGL)}$$

mit $u(T, x) = g(x)$ für $t \in [0, T)$, $x \in \mathbb{R}^d$ mit geeigneten Funktionen $f \colon \mathbb{R}^d \times \mathbb{R} \times \mathbb{R}^d \times \mathbb{R}^{d \times d} \to \mathbb{R}$ und $g \colon \mathbb{R}^d \to \mathbb{R}$. **Ziel:** Näherungsweise lösen von (PDGL). Beispiel: **Wärmeleitungsgleichung**

$$\frac{\partial u}{\partial t}(t, x) + (\Delta_x u)(t, x) = 0.$$

Klassische Näherungsverfahren für (PDGL) leiden unter **Fluch der Dimensionalität**.

Geeignete Modellannahmen liefern:

Funktionaler Zusammenhang zwischen Preise der Basiswerte und Preis des Derivates. **Wertfunktion**

$$u\colon [0, T] \times \mathbb{R}^d \to \mathbb{R}$$

für Derivat mit $d \in \mathbb{N} = \{1, 2, 3, \ldots\}$ Basiswerten. Preis des Derivats

$$u(t, x) \in \mathbb{R}$$

zur Zeit $t \in [0, T]$ in Abhängigkeit der Preise $x = (x_1, \ldots, x_d) \in \mathbb{R}^d$ der Basiswerte. Wertfunktion erfüllt **partielle Differentialgleichung**

$$\frac{\partial u}{\partial t}(t, x) + f\big(x, u(t, x), (\nabla_x u)(t, x), (\operatorname{Hess}_x u)(t, x)\big) = 0 \qquad \text{(PDGL)}$$

mit $u(T, x) = g(x)$ für $t \in [0, T)$, $x \in \mathbb{R}^d$ mit geeigneten Funktionen $f\colon \mathbb{R}^d \times \mathbb{R} \times \mathbb{R}^d \times \mathbb{R}^{d \times d} \to \mathbb{R}$ und $g\colon \mathbb{R}^d \to \mathbb{R}$. **Ziel:** Näherungsweise lösen von (PDGL). Beispiel: **Wärmeleitungsgleichung**

$$\frac{\partial u}{\partial t}(t, x) + (\Delta_x u)(t, x) = 0.$$

Klassische Näherungsverfahren für (PDGL) leiden unter **Fluch der Dimensionalität**.

Geeignete Modellannahmen liefern:

Funktionaler Zusammenhang zwischen Preise der Basiswerte und Preis des Derivates. **Wertfunktion**

$$u\colon [0,T]\times\mathbb{R}^d\to\mathbb{R}$$

für Derivat mit $d\in\mathbb{N}=\{1,2,3,\dots\}$ Basiswerten. Preis des Derivats

$$u(t,x)\in\mathbb{R}$$

zur Zeit $t\in[0,T]$ in Abhängigkeit der Preise $x=(x_1,\dots,x_d)\in\mathbb{R}^d$ der Basiswerte. Wertfunktion erfüllt **partielle Differentialgleichung**

$$\tfrac{\partial u}{\partial t}(t,x)+f\big(x,u(t,x),(\nabla_x u)(t,x),(\operatorname{Hess}_x u)(t,x)\big)=0 \qquad \text{(PDGL)}$$

mit $u(T,x)=g(x)$ für $t\in[0,T)$, $x\in\mathbb{R}^d$ mit geeigneten Funktionen $f\colon\mathbb{R}^d\times\mathbb{R}\times\mathbb{R}^d\times\mathbb{R}^{d\times d}\to\mathbb{R}$ und $g\colon\mathbb{R}^d\to\mathbb{R}$. **Ziel:** Näherungsweise lösen von (PDGL). Beispiel: **Wärmeleitungsgleichung**

$$\tfrac{\partial u}{\partial t}(t,x)+(\Delta_x u)(t,x)=0.$$

Klassische Näherungsverfahren für (PDGL) leiden unter **Fluch der Dimensionalität**.

Geeignete Modellannahmen liefern:

Funktionaler Zusammenhang zwischen Preise der Basiswerte und Preis des Derivates. **Wertfunktion**

$$u\colon [0, T] \times \mathbb{R}^d \to \mathbb{R}$$

für Derivat mit $d \in \mathbb{N} = \{1, 2, 3, \ldots\}$ Basiswerten. Preis des Derivats

$$u(t, x) \in \mathbb{R}$$

zur Zeit $t \in [0, T]$ in Abhängigkeit der Preise $x = (x_1, \ldots, x_d) \in \mathbb{R}^d$ der Basiswerte. Wertfunktion erfüllt **partielle Differentialgleichung**

$$\frac{\partial u}{\partial t}(t, x) + f\big(x, u(t, x), (\nabla_x u)(t, x), (\operatorname{Hess}_x u)(t, x)\big) = 0 \qquad \text{(PDGL)}$$

mit $u(T, x) = g(x)$ für $t \in [0, T)$, $x \in \mathbb{R}^d$ mit geeigneten Funktionen $f\colon \mathbb{R}^d \times \mathbb{R} \times \mathbb{R}^d \times \mathbb{R}^{d \times d} \to \mathbb{R}$ und $g\colon \mathbb{R}^d \to \mathbb{R}$. **Ziel:** Näherungsweise lösen von (PDGL). Beispiel: **Wärmeleitungsgleichung**

$$\frac{\partial u}{\partial t}(t, x) + (\Delta_x u)(t, x) = 0.$$

Klassische Näherungsverfahren für (PDGL) leiden unter **Fluch der Dimensionalität**.

Geeignete Modellannahmen liefern:

Funktionaler Zusammenhang zwischen Preise der Basiswerte und Preis des Derivates. **Wertfunktion**

$$u\colon [0,T] \times \mathbb{R}^d \to \mathbb{R}$$

für Derivat mit $d \in \mathbb{N} = \{1, 2, 3, \ldots\}$ Basiswerten. Preis des Derivats

$$u(t, x) \in \mathbb{R}$$

zur Zeit $t \in [0, T]$ in Abhängigkeit der Preise $x = (x_1, \ldots, x_d) \in \mathbb{R}^d$ der Basiswerte. Wertfunktion erfüllt **partielle Differentialgleichung**

$$\tfrac{\partial u}{\partial t}(t, x) + f\big(x, u(t, x), (\nabla_x u)(t, x), (\operatorname{Hess}_x u)(t, x)\big) = 0 \qquad \text{(PDGL)}$$

mit $u(T, x) = g(x)$ für $t \in [0, T)$, $x \in \mathbb{R}^d$ mit geeigneten Funktionen $f\colon \mathbb{R}^d \times \mathbb{R} \times \mathbb{R}^d \times \mathbb{R}^{d \times d} \to \mathbb{R}$ und $g\colon \mathbb{R}^d \to \mathbb{R}$. **Ziel:** Näherungsweise lösen von (PDGL). Beispiel: **Wärmeleitungsgleichung**

$$\tfrac{\partial u}{\partial t}(t, x) + (\Delta_x u)(t, x) = 0.$$

Klassische Näherungsverfahren für (PDGL) leiden unter **Fluch der Dimensionalität**.

Geeignete Modellannahmen liefern:

Funktionaler Zusammenhang zwischen Preise der Basiswerte und Preis des Derivates. **Wertfunktion**

$$u\colon [0,T] \times \mathbb{R}^d \to \mathbb{R}$$

für Derivat mit $d \in \mathbb{N} = \{1, 2, 3, \dots\}$ Basiswerten. Preis des Derivats

$$u(t, x) \in \mathbb{R}$$

zur Zeit $t \in [0, T]$ in Abhängigkeit der Preise $x = (x_1, \dots, x_d) \in \mathbb{R}^d$ der Basiswerte. Wertfunktion erfüllt **partielle Differentialgleichung**

$$\tfrac{\partial u}{\partial t}(t,x) + f\big(x, u(t,x), (\nabla_x u)(t,x), (\mathrm{Hess}_x\, u)(t,x)\big) = 0 \qquad \text{(PDGL)}$$

mit $u(T, x) = g(x)$ für $t \in [0, T)$, $x \in \mathbb{R}^d$ mit geeigneten Funktionen $f\colon \mathbb{R}^d \times \mathbb{R} \times \mathbb{R}^d \times \mathbb{R}^{d\times d} \to \mathbb{R}$ und $g\colon \mathbb{R}^d \to \mathbb{R}$. **Ziel:** Näherungsweise lösen von (PDGL). Beispiel: **Wärmeleitungsgleichung**

$$\tfrac{\partial u}{\partial t}(t,x) + (\Delta_x u)(t,x) = 0.$$

Klassische Näherungsverfahren für (PDGL) leiden unter **Fluch der Dimensionalität**.

Geeignete Modellannahmen liefern:

Funktionaler Zusammenhang zwischen Preise der Basiswerte und Preis des Derivates. **Wertfunktion**

$$u\colon [0,T]\times\mathbb{R}^d\to\mathbb{R}$$

für Derivat mit $d\in\mathbb{N}=\{1,2,3,\dots\}$ Basiswerten. Preis des Derivats

$$u(t,x)\in\mathbb{R}$$

zur Zeit $t\in[0,T]$ in Abhängigkeit der Preise $x=(x_1,\dots,x_d)\in\mathbb{R}^d$ der Basiswerte. Wertfunktion erfüllt **partielle Differentialgleichung**

$$\tfrac{\partial u}{\partial t}(t,x)+f\big(x,u(t,x),(\nabla_x u)(t,x),(\operatorname{Hess}_x u)(t,x)\big)=0 \qquad \text{(PDGL)}$$

mit $u(T,x)=g(x)$ für $t\in[0,T)$, $x\in\mathbb{R}^d$ mit geeigneten Funktionen $f\colon\mathbb{R}^d\times\mathbb{R}\times\mathbb{R}^d\times\mathbb{R}^{d\times d}\to\mathbb{R}$ und $g\colon\mathbb{R}^d\to\mathbb{R}$. **Ziel:** Näherungsweise lösen von (PDGL). Beispiel: **Wärmeleitungsgleichung**

$$\tfrac{\partial u}{\partial t}(t,x)+(\Delta_x u)(t,x)=0.$$

Klassische Näherungsverfahren für (PDGL) leiden unter **Fluch der Dimensionalität**.

Geeignete Modellannahmen liefern:

Funktionaler Zusammenhang zwischen Preise der Basiswerte und Preis des Derivates. **Wertfunktion**

$$u\colon [0,T]\times\mathbb{R}^d\to\mathbb{R}$$

für Derivat mit $d\in\mathbb{N}=\{1,2,3,\dots\}$ Basiswerten. Preis des Derivats

$$u(t,x)\in\mathbb{R}$$

zur Zeit $t\in[0,T]$ in Abhängigkeit der Preise $x=(x_1,\dots,x_d)\in\mathbb{R}^d$ der Basiswerte. Wertfunktion erfüllt **partielle Differentialgleichung**

$$\frac{\partial u}{\partial t}(t,x)+f\big(x,u(t,x),(\nabla_x u)(t,x),(\mathrm{Hess}_x\, u)(t,x)\big)=0 \qquad \text{(PDGL)}$$

mit $u(T,x)=g(x)$ für $t\in[0,T)$, $x\in\mathbb{R}^d$ mit geeigneten Funktionen $f\colon\mathbb{R}^d\times\mathbb{R}\times\mathbb{R}^d\times\mathbb{R}^{d\times d}\to\mathbb{R}$ und $g\colon\mathbb{R}^d\to\mathbb{R}$. **Ziel:** Näherungsweise lösen von (PDGL). Beispiel: **Wärmeleitungsgleichung**

$$\frac{\partial u}{\partial t}(t,x)+(\Delta_x u)(t,x)=0.$$

Klassische Näherungsverfahren für (PDGL) leiden unter **Fluch der Dimensionalität**.

Geeignete Modellannahmen liefern:

Funktionaler Zusammenhang zwischen Preise der Basiswerte und Preis des Derivates. **Wertfunktion**

$$u\colon [0,T] \times \mathbb{R}^d \to \mathbb{R}$$

für Derivat mit $d \in \mathbb{N} = \{1, 2, 3, \dots\}$ Basiswerten. Preis des Derivats

$$u(t,x) \in \mathbb{R}$$

zur Zeit $t \in [0,T]$ in Abhängigkeit der Preise $x = (x_1, \dots, x_d) \in \mathbb{R}^d$ der Basiswerte. Wertfunktion erfüllt **partielle Differentialgleichung**

$$\frac{\partial u}{\partial t}(t,x) + f\big(x, u(t,x), (\nabla_x u)(t,x), (\operatorname{Hess}_x u)(t,x)\big) = 0 \qquad \text{(PDGL)}$$

mit $u(T,x) = g(x)$ für $t \in [0,T)$, $x \in \mathbb{R}^d$ mit geeigneten Funktionen $f\colon \mathbb{R}^d \times \mathbb{R} \times \mathbb{R}^d \times \mathbb{R}^{d\times d} \to \mathbb{R}$ und $g\colon \mathbb{R}^d \to \mathbb{R}$. **Ziel:** Näherungsweise lösen von (PDGL). Beispiel: **Wärmeleitungsgleichung**

$$\frac{\partial u}{\partial t}(t,x) + (\Delta_x u)(t,x) = 0.$$

Klassische Näherungsverfahren für (PDGL) leiden unter **Fluch der Dimensionalität**.

Geeignete Modellannahmen liefern:

Funktionaler Zusammenhang zwischen Preise der Basiswerte und Preis des Derivates. **Wertfunktion**

$$u\colon [0, T] \times \mathbb{R}^d \to \mathbb{R}$$

für Derivat mit $d \in \mathbb{N} = \{1, 2, 3, \ldots\}$ Basiswerten. Preis des Derivats

$$u(t, x) \in \mathbb{R}$$

zur Zeit $t \in [0, T]$ in Abhängigkeit der Preise $x = (x_1, \ldots, x_d) \in \mathbb{R}^d$ der Basiswerte. Wertfunktion erfüllt **partielle Differentialgleichung**

$$\tfrac{\partial u}{\partial t}(t, x) + f\big(x, u(t, x), (\nabla_x u)(t, x), (\mathrm{Hess}_x\, u)(t, x)\big) = 0 \qquad \text{(PDGL)}$$

mit $u(T, x) = g(x)$ für $t \in [0, T)$, $x \in \mathbb{R}^d$ mit geeigneten Funktionen $f\colon \mathbb{R}^d \times \mathbb{R} \times \mathbb{R}^d \times \mathbb{R}^{d \times d} \to \mathbb{R}$ und $g\colon \mathbb{R}^d \to \mathbb{R}$. **Ziel:** Näherungsweise lösen von (PDGL). Beispiel: **Wärmeleitungsgleichung**

$$\tfrac{\partial u}{\partial t}(t, x) + (\Delta_x u)(t, x) = 0.$$

Klassische Näherungsverfahren für (PDGL) leiden unter **Fluch der Dimensionalität**.

Geeignete Modellannahmen liefern:

Funktionaler Zusammenhang zwischen Preise der Basiswerte und Preis des Derivates. **Wertfunktion**

$$u\colon [0,T] \times \mathbb{R}^d \to \mathbb{R}$$

für Derivat mit $d \in \mathbb{N} = \{1, 2, 3, \dots\}$ Basiswerten. Preis des Derivats

$$u(t, x) \in \mathbb{R}$$

zur Zeit $t \in [0, T]$ in Abhängigkeit der Preise $x = (x_1, \dots, x_d) \in \mathbb{R}^d$ der Basiswerte. Wertfunktion erfüllt **partielle Differentialgleichung**

$$\frac{\partial u}{\partial t}(t, x) + f\big(x, u(t, x), (\nabla_x u)(t, x), (\mathrm{Hess}_x\, u)(t, x)\big) = 0 \qquad \text{(PDGL)}$$

mit $u(T, x) = g(x)$ für $t \in [0, T)$, $x \in \mathbb{R}^d$ mit geeigneten Funktionen $f\colon \mathbb{R}^d \times \mathbb{R} \times \mathbb{R}^d \times \mathbb{R}^{d \times d} \to \mathbb{R}$ und $g\colon \mathbb{R}^d \to \mathbb{R}$. **Ziel:** Näherungsweise lösen von (PDGL). Beispiel: **Wärmeleitungsgleichung**

$$\frac{\partial u}{\partial t}(t, x) + (\Delta_x u)(t, x) = 0.$$

Klassische Näherungsverfahren für (PDGL) leiden unter **Fluch der Dimensionalität**.

Beispiel: **Wärmeleitungsgleichung**

$$\frac{\partial u}{\partial t}(t,x) + (\Delta_x u)(t,x) = 0.$$

Klassische Näherungsverfahren für (PDGL) leiden unter dem **Fluch der Dimensionalität**.

Finite-Differenzen-Methode: Betrachte $N \in \mathbb{N}$, Gitterpunkte

$$x_m = \left(\frac{m_1}{N}, \frac{m_2}{N}, \dots, \frac{m_d}{N}\right)$$

für $m = (m_1, \dots, m_d) \in \mathbb{Z}^d$ (*Räumliche Diskretisierung*) und

$$t_n = \frac{nT}{N}$$

für $n \in \{0, 1, 2, \dots, N\}$ (*Zeitliche Diskretisierung*), Näherungen

$$\mathbb{U}_{n,m} \approx u(t_n, x_m) \in \mathbb{R}$$

und *Differenzen-Quotienten* in (PDGL). Diskretisierungsfehler $\exists\, C > 0 \colon \forall\, N \in \mathbb{N} \colon$

$$\max_{m \in \{0,1,\dots,N\}^d} \left|u(t_0, x_m) - \mathbb{U}_{0,m}\right| \leq C \cdot N^{-1}$$

Rechenaufwand versus **Diskretisierungsfehler**:
N^{d+1} Gitterpunkte versus Diskretisierungsfehler $\varepsilon = C \cdot N^{-1}$.
Also $C^{d+1}\varepsilon^{-(d+1)}$ Gitterpunkte. Rechenaufwand wächst **exponentiell** in d.

Beispiel: **Wärmeleitungsgleichung**

$$\frac{\partial u}{\partial t}(t,x) + (\Delta_x u)(t,x) = 0.$$

Klassische Näherungsverfahren für (PDGL) leiden unter dem **Fluch der Dimensionalität**.

Finite-Differenzen-Methode: Betrachte $N \in \mathbb{N}$, Gitterpunkte

$$x_m = \left(\frac{m_1}{N}, \frac{m_2}{N}, \ldots, \frac{m_d}{N}\right)$$

für $m = (m_1, \ldots, m_d) \in \mathbb{Z}^d$ (*Räumliche Diskretisierung*) und

$$t_n = \frac{nT}{N}$$

für $n \in \{0, 1, 2, \ldots, N\}$ (*Zeitliche Diskretisierung*), Näherungen

$$\mathbb{U}_{n,m} \approx u(t_n, x_m) \in \mathbb{R}$$

und *Differenzen-Quotienten* in (PDGL). Diskretisierungsfehler $\exists\, C > 0 : \forall\, N \in \mathbb{N}:$

$$\max_{m \in \{0,1,\ldots,N\}^d} |u(t_0, x_m) - \mathbb{U}_{0,m}| \leq C \cdot N^{-1}$$

Rechenaufwand versus **Diskretisierungsfehler**:
N^{d+1} Gitterpunkte versus Diskretisierungsfehler $\varepsilon = C \cdot N^{-1}$.
Also $C^{d+1}\varepsilon^{-(d+1)}$ Gitterpunkte. Rechenaufwand wächst **exponentiell** in d.

Beispiel: **Wärmeleitungsgleichung**

$$\frac{\partial u}{\partial t}(t,x) + (\Delta_x u)(t,x) = 0.$$

Klassische Näherungsverfahren für (PDGL) leiden unter dem **Fluch der Dimensionalität**.

Finite-Differenzen-Methode: Betrachte $N \in \mathbb{N}$, Gitterpunkte

$$x_m = \left(\frac{m_1}{N}, \frac{m_2}{N}, \ldots, \frac{m_d}{N}\right)$$

für $m = (m_1, \ldots, m_d) \in \mathbb{Z}^d$ (*Räumliche Diskretisierung*) und

$$t_n = \frac{nT}{N}$$

für $n \in \{0, 1, 2, \ldots, N\}$ (*Zeitliche Diskretisierung*), Näherungen

$$\mathbb{U}_{n,m} \approx u(t_n, x_m) \in \mathbb{R}$$

und *Differenzen-Quotienten* in (PDGL). Diskretisierungsfehler $\exists\, C > 0 \colon \forall\, N \in \mathbb{N} \colon$

$$\max_{m \in \{0,1,\ldots,N\}^d} \left|u(t_0, x_m) - \mathbb{U}_{0,m}\right| \le C \cdot N^{-1}$$

Rechenaufwand versus **Diskretisierungsfehler**:
N^{d+1} Gitterpunkte versus Diskretisierungsfehler $\varepsilon = C \cdot N^{-1}$.
Also $C^{d+1}\varepsilon^{-(d+1)}$ Gitterpunkte. Rechenaufwand wächst **exponentiell** in d.

Beispiel: **Wärmeleitungsgleichung**

$$\frac{\partial u}{\partial t}(t,x) + (\Delta_x u)(t,x) = 0.$$

Klassische Näherungsverfahren für (PDGL) leiden unter dem **Fluch der Dimensionalität**.

Finite-Differenzen-Methode: Betrachte $N \in \mathbb{N}$, Gitterpunkte

$$x_m = \left(\frac{m_1}{N}, \frac{m_2}{N}, \dots, \frac{m_d}{N}\right)$$

für $m = (m_1, \dots, m_d) \in \mathbb{Z}^d$ (*Räumliche Diskretisierung*) und

$$t_n = \frac{nT}{N}$$

für $n \in \{0, 1, 2, \dots, N\}$ (*Zeitliche Diskretisierung*), Näherungen

$$\mathbb{U}_{n,m} \approx u(t_n, x_m) \in \mathbb{R}$$

und *Differenzen-Quotienten* in (PDGL). Diskretisierungsfehler $\exists\, C > 0 \colon \forall\, N \in \mathbb{N}$:

$$\max_{m \in \{0,1,\dots,N\}^d} \left|u(t_0, x_m) - \mathbb{U}_{0,m}\right| \leq C \cdot N^{-1}$$

Rechenaufwand versus **Diskretisierungsfehler**:

N^{d+1} Gitterpunkte versus Diskretisierungsfehler $\varepsilon = C \cdot N^{-1}$.

Also $C^{d+1}\varepsilon^{-(d+1)}$ Gitterpunkte. Rechenaufwand wächst **exponentiell** in d.

Beispiel: **Wärmeleitungsgleichung**

$$\frac{\partial u}{\partial t}(t,x) + (\Delta_x u)(t,x) = 0.$$

Klassische Näherungsverfahren für (PDGL) leiden unter dem **Fluch der Dimensionalität**.

Finite-Differenzen-Methode: Betrachte $N \in \mathbb{N}$, Gitterpunkte

$$x_m = \left(\frac{m_1}{N}, \frac{m_2}{N}, \ldots, \frac{m_d}{N}\right)$$

für $m = (m_1, \ldots, m_d) \in \mathbb{Z}^d$ (*Räumliche Diskretisierung*) und

$$t_n = \frac{nT}{N}$$

für $n \in \{0, 1, 2, \ldots, N\}$ (*Zeitliche Diskretisierung*), Näherungen

$$\mathbb{U}_{n,m} \approx u(t_n, x_m) \in \mathbb{R}$$

und *Differenzen-Quotienten* in (PDGL). Diskretisierungsfehler $\exists\, C > 0 : \forall\, N \in \mathbb{N}:$

$$\max_{m \in \{0,1,\ldots,N\}^d} \left|u(t_0, x_m) - \mathbb{U}_{0,m}\right| \leq C \cdot N^{-1}$$

Rechenaufwand versus **Diskretisierungsfehler**:

N^{d+1} Gitterpunkte versus Diskretisierungsfehler $\varepsilon = C \cdot N^{-1}$.

Also $C^{d+1}\varepsilon^{-(d+1)}$ Gitterpunkte. Rechenaufwand wächst **exponentiell** in d.

Beispiel: **Wärmeleitungsgleichung**

$$\frac{\partial u}{\partial t}(t,x) + (\Delta_x u)(t,x) = 0.$$

Klassische Näherungsverfahren für (PDGL) leiden unter dem **Fluch der Dimensionalität**.

Finite-Differenzen-Methode: Betrachte $N \in \mathbb{N}$, Gitterpunkte

$$x_m = \left(\frac{m_1}{N}, \frac{m_2}{N}, \ldots, \frac{m_d}{N}\right)$$

für $m = (m_1, \ldots, m_d) \in \mathbb{Z}^d$ (*Räumliche Diskretisierung*) und

$$t_n = \frac{nT}{N}$$

für $n \in \{0, 1, 2, \ldots, N\}$ (*Zeitliche Diskretisierung*), Näherungen

$$\mathbb{U}_{n,m} \approx u(t_n, x_m) \in \mathbb{R}$$

und *Differenzen-Quotienten* in (PDGL). Diskretisierungsfehler $\exists\, C > 0 \colon \forall\, N \in \mathbb{N}\colon$

$$\max_{m \in \{0,1,\ldots,N\}^d} \left|u(t_0, x_m) - \mathbb{U}_{0,m}\right| \le C \cdot N^{-1}$$

Rechenaufwand versus **Diskretisierungsfehler**:

N^{d+1} Gitterpunkte versus Diskretisierungsfehler $\varepsilon = C \cdot N^{-1}$.

Also $C^{d+1}\varepsilon^{-(d+1)}$ Gitterpunkte. Rechenaufwand wächst **exponentiell** in d.

Beispiel: **Wärmeleitungsgleichung**

$$\frac{\partial u}{\partial t}(t,x) + (\Delta_x u)(t,x) = 0.$$

Klassische Näherungsverfahren für (PDGL) leiden unter dem **Fluch der Dimensionalität**.

Finite-Differenzen-Methode: Betrachte $N \in \mathbb{N}$, Gitterpunkte

$$x_m = \left(\frac{m_1}{N}, \frac{m_2}{N}, \dots, \frac{m_d}{N}\right)$$

für $m = (m_1, \dots, m_d) \in \mathbb{Z}^d$ (*Räumliche Diskretisierung*) und

$$t_n = \frac{nT}{N}$$

für $n \in \{0, 1, 2, \dots, N\}$ (*Zeitliche Diskretisierung*), Näherungen

$$\mathbb{U}_{n,m} \approx u(t_n, x_m) \in \mathbb{R}$$

und *Differenzen-Quotienten* in (PDGL). Diskretisierungsfehler $\exists\, C > 0 \colon \forall\, N \in \mathbb{N}\colon$

$$\max_{m \in \{0,1,\dots,N\}^d} \left|u(t_0, x_m) - \mathbb{U}_{0,m}\right| \leq C \cdot N^{-1}$$

Rechenaufwand versus **Diskretisierungsfehler**:

N^{d+1} Gitterpunkte versus Diskretisierungsfehler $\varepsilon = C \cdot N^{-1}$.

Also $C^{d+1}\varepsilon^{-(d+1)}$ Gitterpunkte. Rechenaufwand wächst **exponentiell** in d.

Beispiel: **Wärmeleitungsgleichung**

$$\frac{\partial u}{\partial t}(t,x) + (\Delta_x u)(t,x) = 0.$$

Klassische Näherungsverfahren für (PDGL) leiden unter dem **Fluch der Dimensionalität**.

Finite-Differenzen-Methode: Betrachte $N \in \mathbb{N}$, Gitterpunkte

$$x_m = \left(\frac{m_1}{N}, \frac{m_2}{N}, \ldots, \frac{m_d}{N}\right)$$

für $m = (m_1, \ldots, m_d) \in \mathbb{Z}^d$ (*Räumliche Diskretisierung*) und

$$t_n = \frac{nT}{N}$$

für $n \in \{0, 1, 2, \ldots, N\}$ (*Zeitliche Diskretisierung*), Näherungen

$$\mathbb{U}_{n,m} \approx u(t_n, x_m) \in \mathbb{R}$$

und *Differenzen-Quotienten* in (PDGL). Diskretisierungsfehler $\exists\, C > 0 \colon \forall\, N \in \mathbb{N}\colon$

$$\max_{m \in \{0,1,\ldots,N\}^d} \left|u(t_0, x_m) - \mathbb{U}_{0,m}\right| \leq C \cdot N^{-1}$$

Rechenaufwand versus **Diskretisierungsfehler**:

N^{d+1} Gitterpunkte versus Diskretisierungsfehler $\varepsilon = C \cdot N^{-1}$.

Also $C^{d+1}\varepsilon^{-(d+1)}$ Gitterpunkte. Rechenaufwand wächst **exponentiell** in d.

Beispiel: **Wärmeleitungsgleichung**

$$\frac{\partial u}{\partial t}(t,x) + (\Delta_x u)(t,x) = 0.$$

Klassische Näherungsverfahren für (PDGL) leiden unter dem **Fluch der Dimensionalität**.

Finite-Differenzen-Methode: Betrachte $N \in \mathbb{N}$, Gitterpunkte

$$x_m = \left(\frac{m_1}{N}, \frac{m_2}{N}, \ldots, \frac{m_d}{N}\right)$$

für $m = (m_1, \ldots, m_d) \in \mathbb{Z}^d$ (*Räumliche Diskretisierung*) und

$$t_n = \frac{nT}{N}$$

für $n \in \{0, 1, 2, \ldots, N\}$ (*Zeitliche Diskretisierung*), Näherungen

$$\mathbb{U}_{n,m} \approx u(t_n, x_m) \in \mathbb{R}$$

und *Differenzen-Quotienten* in (PDGL). Diskretisierungsfehler $\exists\, C > 0 \colon \forall\, N \in \mathbb{N}\colon$

$$\max_{m \in \{0,1,\ldots,N\}^d} \left|u(t_0, x_m) - \mathbb{U}_{0,m}\right| \leq C \cdot N^{-1}$$

Rechenaufwand versus **Diskretisierungsfehler**:

N^{d+1} Gitterpunkte versus Diskretisierungsfehler $\varepsilon = C \cdot N^{-1}$.

Also $C^{d+1}\varepsilon^{-(d+1)}$ Gitterpunkte. Rechenaufwand wächst **exponentiell** in d.

Beispiel: **Wärmeleitungsgleichung**

$$\frac{\partial u}{\partial t}(t,x) + (\Delta_x u)(t,x) = 0.$$

Klassische Näherungsverfahren für (PDGL) leiden unter dem **Fluch der Dimensionalität**.

Finite-Differenzen-Methode: Betrachte $N \in \mathbb{N}$, Gitterpunkte

$$x_m = \left(\tfrac{m_1}{N}, \tfrac{m_2}{N}, \ldots, \tfrac{m_d}{N}\right)$$

für $m = (m_1, \ldots, m_d) \in \mathbb{Z}^d$ (*Räumliche Diskretisierung*) und

$$t_n = \tfrac{nT}{N}$$

für $n \in \{0, 1, 2, \ldots, N\}$ (*Zeitliche Diskretisierung*), Näherungen

$$\mathbb{U}_{n,m} \approx u(t_n, x_m) \in \mathbb{R}$$

und *Differenzen-Quotienten* in (PDGL). Diskretisierungsfehler $\exists\, C > 0\colon \forall\, N \in \mathbb{N}\colon$

$$\max_{m \in \{0,1,\ldots,N\}^d} \left|u(t_0, x_m) - \mathbb{U}_{0,m}\right| \leq C \cdot N^{-1}$$

Rechenaufwand versus **Diskretisierungsfehler**:
N^{d+1} Gitterpunkte versus Diskretisierungsfehler $\varepsilon = C \cdot N^{-1}$.
Also $C^{d+1}\varepsilon^{-(d+1)}$ Gitterpunkte. Rechenaufwand wächst **exponentiell** in d.

Künstliches neuronales Netz mit $s \in \mathbb{N}$ verborgenen Schichten

- $\rho \in \mathbb{N}$ hinreichend groß,
- Funktion $\mathcal{R}\colon \mathbb{R}^d \to \mathbb{R}^d$ mit $\forall\, x = (x_1, \ldots, x_d) \in \mathbb{R}^d$:

$$\mathcal{R}(x) = (\max\{x_1, 0\}, \ldots, \max\{x_d, 0\}),$$

- $\forall\, \theta = (\theta_1, \ldots, \theta_\rho) \in \mathbb{R}^\rho$, $v \in \mathbb{N}_0$, $k, l \in \mathbb{N}$ mit $v + k(l+1) \leq \rho$ Funktion $A^{\theta,v}_{k,l}\colon \mathbb{R}^l \to \mathbb{R}^k$ mit $\forall\, x = (x_1, \ldots, x_l) \in \mathbb{R}^l$:

$$A^{\theta,v}_{k,l}(x) = \begin{pmatrix} \theta_{v+1} & \theta_{v+2} & \ldots & \theta_{v+l} \\ \theta_{v+l+1} & \theta_{v+l+2} & \ldots & \theta_{v+2l} \\ \vdots & \vdots & \vdots & \vdots \\ \theta_{v+(k-1)l+1} & \theta_{v+(k-1)l+2} & \ldots & \theta_{v+kl} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_l \end{pmatrix} + \begin{pmatrix} \theta_{v+kl+1} \\ \theta_{v+kl+2} \\ \vdots \\ \theta_{v+kl+k} \end{pmatrix},$$

- $\forall\, \theta \in \mathbb{R}^\rho$ Funktion $\mathcal{U}_\theta\colon \mathbb{R}^d \to \mathbb{R}$ mit

$$\mathcal{U}_\theta = A^{\theta,sd(d+1)}_{1,d} \circ \mathcal{R} \circ A^{\theta,(s-1)d(d+1)}_{d,d} \circ \ldots \circ \mathcal{R} \circ A^{\theta,d(d+1)}_{d,d} \circ \mathcal{R} \circ A^{\theta,0}_{d,d}.$$

Für $x \in \mathbb{R}^d$ und geeignete $\theta \in \mathbb{R}^\rho$ betrachten wir die Näherung $\mathcal{U}_\theta(x) \approx u(0, x) \in \mathbb{R}$.

Künstliches neuronales Netz mit $s \in \mathbb{N}$ verborgenen Schichten

- $\rho \in \mathbb{N}$ hinreichend groß,
- Funktion $\mathcal{R}\colon \mathbb{R}^d \to \mathbb{R}^d$ mit $\forall\, x = (x_1, \ldots, x_d) \in \mathbb{R}^d$:

$$\mathcal{R}(x) = (\max\{x_1, 0\}, \ldots, \max\{x_d, 0\}),$$

- $\forall\, \theta = (\theta_1, \ldots, \theta_\rho) \in \mathbb{R}^\rho$, $v \in \mathbb{N}_0$, $k, l \in \mathbb{N}$ mit $v + k(l+1) \leq \rho$ Funktion $A^{\theta,v}_{k,l}\colon \mathbb{R}^l \to \mathbb{R}^k$ mit $\forall\, x = (x_1, \ldots, x_l) \in \mathbb{R}^l$:

$$A^{\theta,v}_{k,l}(x) = \begin{pmatrix} \theta_{v+1} & \theta_{v+2} & \ldots & \theta_{v+l} \\ \theta_{v+l+1} & \theta_{v+l+2} & \ldots & \theta_{v+2l} \\ \vdots & \vdots & \vdots & \vdots \\ \theta_{v+(k-1)l+1} & \theta_{v+(k-1)l+2} & \ldots & \theta_{v+kl} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_l \end{pmatrix} + \begin{pmatrix} \theta_{v+kl+1} \\ \theta_{v+kl+2} \\ \vdots \\ \theta_{v+kl+k} \end{pmatrix},$$

- $\forall\, \theta \in \mathbb{R}^\rho$ Funktion $\mathcal{U}_\theta\colon \mathbb{R}^d \to \mathbb{R}$ mit

$$\mathcal{U}_\theta = A^{\theta,sd(d+1)}_{1,d} \circ \mathcal{R} \circ A^{\theta,(s-1)d(d+1)}_{d,d} \circ \ldots \circ \mathcal{R} \circ A^{\theta,d(d+1)}_{d,d} \circ \mathcal{R} \circ A^{\theta,0}_{d,d}.$$

Für $x \in \mathbb{R}^d$ und geeignete $\theta \in \mathbb{R}^\rho$ betrachten wir die Näherung $\mathcal{U}_\theta(x) \approx u(0, x) \in \mathbb{R}$.

Künstliches neuronales Netz mit $s \in \mathbb{N}$ verborgenen Schichten

- $\rho \in \mathbb{N}$ hinreichend groß,
- Funktion $\mathcal{R}\colon \mathbb{R}^d \to \mathbb{R}^d$ mit $\forall\, x = (x_1, \ldots, x_d) \in \mathbb{R}^d$:

$$\mathcal{R}(x) = (\max\{x_1, 0\}, \ldots, \max\{x_d, 0\}),$$

- $\forall\, \theta = (\theta_1, \ldots, \theta_\rho) \in \mathbb{R}^\rho$, $v \in \mathbb{N}_0$, $k, l \in \mathbb{N}$ mit $v + k(l+1) \leq \rho$ Funktion $A_{k,l}^{\theta,v}\colon \mathbb{R}^l \to \mathbb{R}^k$ mit $\forall\, x = (x_1, \ldots, x_l) \in \mathbb{R}^l$:

$$A_{k,l}^{\theta,v}(x) = \begin{pmatrix} \theta_{v+1} & \theta_{v+2} & \cdots & \theta_{v+l} \\ \theta_{v+l+1} & \theta_{v+l+2} & \cdots & \theta_{v+2l} \\ \vdots & \vdots & \vdots & \vdots \\ \theta_{v+(k-1)l+1} & \theta_{v+(k-1)l+2} & \cdots & \theta_{v+kl} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_l \end{pmatrix} + \begin{pmatrix} \theta_{v+kl+1} \\ \theta_{v+kl+2} \\ \vdots \\ \theta_{v+kl+k} \end{pmatrix},$$

- $\forall\, \theta \in \mathbb{R}^\rho$ Funktion $\mathcal{U}_\theta\colon \mathbb{R}^d \to \mathbb{R}$ mit

$$\mathcal{U}_\theta = A_{1,d}^{\theta,sd(d+1)} \circ \mathcal{R} \circ A_{d,d}^{\theta,(s-1)d(d+1)} \circ \ldots \circ \mathcal{R} \circ A_{d,d}^{\theta,d(d+1)} \circ \mathcal{R} \circ A_{d,d}^{\theta,0}.$$

Für $x \in \mathbb{R}^d$ und geeignete $\theta \in \mathbb{R}^\rho$ betrachten wir die Näherung $\mathcal{U}_\theta(x) \approx u(0, x) \in \mathbb{R}$.

Künstliches neuronales Netz mit $s \in \mathbb{N}$ verborgenen Schichten

- $\rho \in \mathbb{N}$ hinreichend groß,
- Funktion $\mathcal{R} \colon \mathbb{R}^d \to \mathbb{R}^d$ mit $\forall\, x = (x_1, \ldots, x_d) \in \mathbb{R}^d$:

$$\mathcal{R}(x) = \left(\max\{x_1, 0\}, \ldots, \max\{x_d, 0\}\right),$$

- $\forall\, \theta = (\theta_1, \ldots, \theta_\rho) \in \mathbb{R}^\rho$, $v \in \mathbb{N}_0$, $k, l \in \mathbb{N}$ mit $v + k(l+1) \leq \rho$ Funktion $A^{\theta,v}_{k,l} \colon \mathbb{R}^l \to \mathbb{R}^k$ mit $\forall\, x = (x_1, \ldots, x_l) \in \mathbb{R}^l$:

$$A^{\theta,v}_{k,l}(x) = \begin{pmatrix} \theta_{v+1} & \theta_{v+2} & \cdots & \theta_{v+l} \\ \theta_{v+l+1} & \theta_{v+l+2} & \cdots & \theta_{v+2l} \\ \vdots & \vdots & \vdots & \vdots \\ \theta_{v+(k-1)l+1} & \theta_{v+(k-1)l+2} & \cdots & \theta_{v+kl} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_l \end{pmatrix} + \begin{pmatrix} \theta_{v+kl+1} \\ \theta_{v+kl+2} \\ \vdots \\ \theta_{v+kl+k} \end{pmatrix},$$

- $\forall\, \theta \in \mathbb{R}^\rho$ Funktion $\mathcal{U}_\theta \colon \mathbb{R}^d \to \mathbb{R}$ mit

$$\mathcal{U}_\theta = A^{\theta,sd(d+1)}_{1,d} \circ \mathcal{R} \circ A^{\theta,(s-1)d(d+1)}_{d,d} \circ \ldots \circ \mathcal{R} \circ A^{\theta,d(d+1)}_{d,d} \circ \mathcal{R} \circ A^{\theta,0}_{d,d}.$$

Für $x \in \mathbb{R}^d$ und geeignete $\theta \in \mathbb{R}^\rho$ betrachten wir die Näherung $\mathcal{U}_\theta(x) \approx u(0,x) \in \mathbb{R}$.

Künstliches neuronales Netz mit $s \in \mathbb{N}$ verborgenen Schichten

- $\rho \in \mathbb{N}$ hinreichend groß,
- Funktion $\mathcal{R} \colon \mathbb{R}^d \to \mathbb{R}^d$ mit $\forall\, x = (x_1, \dots, x_d) \in \mathbb{R}^d$:

$$\mathcal{R}(x) = \left(\max\{x_1, 0\}, \dots, \max\{x_d, 0\}\right),$$

- $\forall\, \theta = (\theta_1, \dots, \theta_\rho) \in \mathbb{R}^\rho$, $v \in \mathbb{N}_0$, $k, l \in \mathbb{N}$ mit $v + k(l+1) \le \rho$ Funktion $A^{\theta,v}_{k,l} \colon \mathbb{R}^l \to \mathbb{R}^k$ mit $\forall\, x = (x_1, \dots, x_l) \in \mathbb{R}^l$:

$$A^{\theta,v}_{k,l}(x) = \begin{pmatrix} \theta_{v+1} & \theta_{v+2} & \dots & \theta_{v+l} \\ \theta_{v+l+1} & \theta_{v+l+2} & \dots & \theta_{v+2l} \\ \vdots & \vdots & \vdots & \vdots \\ \theta_{v+(k-1)l+1} & \theta_{v+(k-1)l+2} & \dots & \theta_{v+kl} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_l \end{pmatrix} + \begin{pmatrix} \theta_{v+kl+1} \\ \theta_{v+kl+2} \\ \vdots \\ \theta_{v+kl+k} \end{pmatrix},$$

- $\forall\, \theta \in \mathbb{R}^\rho$ Funktion $\mathcal{U}_\theta \colon \mathbb{R}^d \to \mathbb{R}$ mit

$$\mathcal{U}_\theta = A^{\theta,sd(d+1)}_{1,d} \circ \mathcal{R} \circ A^{\theta,(s-1)d(d+1)}_{d,d} \circ \dots \circ \mathcal{R} \circ A^{\theta,d(d+1)}_{d,d} \circ \mathcal{R} \circ A^{\theta,0}_{d,d}.$$

Für $x \in \mathbb{R}^d$ und geeignete $\theta \in \mathbb{R}^\rho$ betrachten wir die Näherung $\mathcal{U}_\theta(x) \approx u(0,x) \in \mathbb{R}$.

Künstliches neuronales Netz mit $s \in \mathbb{N}$ verborgenen Schichten

- $\rho \in \mathbb{N}$ hinreichend groß,
- Funktion $\mathcal{R} \colon \mathbb{R}^d \to \mathbb{R}^d$ mit $\forall\, x = (x_1, \ldots, x_d) \in \mathbb{R}^d$:

$$\mathcal{R}(x) = \left(\max\{x_1, 0\}, \ldots, \max\{x_d, 0\}\right),$$

- $\forall\, \theta = (\theta_1, \ldots, \theta_\rho) \in \mathbb{R}^\rho$, $v \in \mathbb{N}_0$, $k, l \in \mathbb{N}$ mit $v + k(l+1) \leq \rho$ Funktion $A_{k,l}^{\theta,v} \colon \mathbb{R}^l \to \mathbb{R}^k$ mit $\forall\, x = (x_1, \ldots, x_l) \in \mathbb{R}^l$:

$$A_{k,l}^{\theta,v}(x) = \begin{pmatrix} \theta_{v+1} & \theta_{v+2} & \cdots & \theta_{v+l} \\ \theta_{v+l+1} & \theta_{v+l+2} & \cdots & \theta_{v+2l} \\ \vdots & \vdots & \vdots & \vdots \\ \theta_{v+(k-1)l+1} & \theta_{v+(k-1)l+2} & \cdots & \theta_{v+kl} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_l \end{pmatrix} + \begin{pmatrix} \theta_{v+kl+1} \\ \theta_{v+kl+2} \\ \vdots \\ \theta_{v+kl+k} \end{pmatrix},$$

- $\forall\, \theta \in \mathbb{R}^\rho$ Funktion $\mathcal{U}_\theta \colon \mathbb{R}^d \to \mathbb{R}$ mit

$$\mathcal{U}_\theta = A_{1,d}^{\theta, sd(d+1)} \circ \mathcal{R} \circ A_{d,d}^{\theta,(s-1)d(d+1)} \circ \ldots \circ \mathcal{R} \circ A_{d,d}^{\theta,d(d+1)} \circ \mathcal{R} \circ A_{d,d}^{\theta,0}.$$

Für $x \in \mathbb{R}^d$ und geeignete $\theta \in \mathbb{R}^\rho$ betrachten wir die Näherung $\mathcal{U}_\theta(x) \approx u(0,x) \in \mathbb{R}$.

Künstliches neuronales Netz mit $s \in \mathbb{N}$ verborgenen Schichten

- $\rho \in \mathbb{N}$ hinreichend groß,
- Funktion $\mathcal{R} \colon \mathbb{R}^d \to \mathbb{R}^d$ mit $\forall\, x = (x_1, \ldots, x_d) \in \mathbb{R}^d$:

$$\mathcal{R}(x) = \left(\max\{x_1, 0\}, \ldots, \max\{x_d, 0\}\right),$$

- $\forall\, \theta = (\theta_1, \ldots, \theta_\rho) \in \mathbb{R}^\rho$, $v \in \mathbb{N}_0$, $k, l \in \mathbb{N}$ mit $v + k(l+1) \le \rho$ Funktion $A_{k,l}^{\theta,v} \colon \mathbb{R}^l \to \mathbb{R}^k$ mit $\forall\, x = (x_1, \ldots, x_l) \in \mathbb{R}^l$:

$$A_{k,l}^{\theta,v}(x) = \begin{pmatrix} \theta_{v+1} & \theta_{v+2} & \ldots & \theta_{v+l} \\ \theta_{v+l+1} & \theta_{v+l+2} & \ldots & \theta_{v+2l} \\ \vdots & \vdots & \vdots & \vdots \\ \theta_{v+(k-1)l+1} & \theta_{v+(k-1)l+2} & \ldots & \theta_{v+kl} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_l \end{pmatrix} + \begin{pmatrix} \theta_{v+kl+1} \\ \theta_{v+kl+2} \\ \vdots \\ \theta_{v+kl+k} \end{pmatrix},$$

- $\forall\, \theta \in \mathbb{R}^\rho$ Funktion $\mathcal{U}_\theta \colon \mathbb{R}^d \to \mathbb{R}$ mit

$$\mathcal{U}_\theta = A_{1,d}^{\theta, sd(d+1)} \circ \mathcal{R} \circ A_{d,d}^{\theta,(s-1)d(d+1)} \circ \ldots \circ \mathcal{R} \circ A_{d,d}^{\theta,d(d+1)} \circ \mathcal{R} \circ A_{d,d}^{\theta,0}.$$

Für $x \in \mathbb{R}^d$ und geeignete $\theta \in \mathbb{R}^\rho$ betrachten wir die Näherung $\mathcal{U}_\theta(x) \approx u(0,x) \in \mathbb{R}$.

Künstliches neuronales Netz mit $s \in \mathbb{N}$ verborgenen Schichten

- $\rho \in \mathbb{N}$ hinreichend groß,
- Funktion $\mathcal{R} \colon \mathbb{R}^d \to \mathbb{R}^d$ mit $\forall\, x = (x_1, \ldots, x_d) \in \mathbb{R}^d$:

$$\mathcal{R}(x) = (\max\{x_1, 0\}, \ldots, \max\{x_d, 0\}),$$

- $\forall\, \theta = (\theta_1, \ldots, \theta_\rho) \in \mathbb{R}^\rho$, $v \in \mathbb{N}_0$, $k, l \in \mathbb{N}$ mit $v + k(l+1) \le \rho$ Funktion $A^{\theta,v}_{k,l} \colon \mathbb{R}^l \to \mathbb{R}^k$ mit $\forall\, x = (x_1, \ldots, x_l) \in \mathbb{R}^l$:

$$A^{\theta,v}_{k,l}(x) = \begin{pmatrix} \theta_{v+1} & \theta_{v+2} & \cdots & \theta_{v+l} \\ \theta_{v+l+1} & \theta_{v+l+2} & \cdots & \theta_{v+2l} \\ \vdots & \vdots & \vdots & \vdots \\ \theta_{v+(k-1)l+1} & \theta_{v+(k-1)l+2} & \cdots & \theta_{v+kl} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_l \end{pmatrix} + \begin{pmatrix} \theta_{v+kl+1} \\ \theta_{v+kl+2} \\ \vdots \\ \theta_{v+kl+k} \end{pmatrix},$$

- $\forall\, \theta \in \mathbb{R}^\rho$ Funktion $\mathcal{U}_\theta \colon \mathbb{R}^d \to \mathbb{R}$ mit

$$\mathcal{U}_\theta = A^{\theta,sd(d+1)}_{1,d} \circ \mathcal{R} \circ A^{\theta,(s-1)d(d+1)}_{d,d} \circ \ldots \circ \mathcal{R} \circ A^{\theta,d(d+1)}_{d,d} \circ \mathcal{R} \circ A^{\theta,0}_{d,d}.$$

Für $x \in \mathbb{R}^d$ und geeignete $\theta \in \mathbb{R}^\rho$ betrachten wir die Näherung
$\mathcal{U}_\theta(x) \approx u(0, x) \in \mathbb{R}$.

Künstliches neuronales Netz mit $s \in \mathbb{N}$ verborgenen Schichten

- $\rho \in \mathbb{N}$ hinreichend groß,
- Funktion $\mathcal{R}\colon \mathbb{R}^d \to \mathbb{R}^d$ mit $\forall\, x = (x_1, \dots, x_d) \in \mathbb{R}^d$:

$$\mathcal{R}(x) = \left(\max\{x_1, 0\}, \dots, \max\{x_d, 0\}\right),$$

- $\forall\, \theta = (\theta_1, \dots, \theta_\rho) \in \mathbb{R}^\rho$, $v \in \mathbb{N}_0$, $k, l \in \mathbb{N}$ mit $v + k(l+1) \leq \rho$ Funktion $A_{k,l}^{\theta,v}\colon \mathbb{R}^l \to \mathbb{R}^k$ mit $\forall\, x = (x_1, \dots, x_l) \in \mathbb{R}^l$:

$$A_{k,l}^{\theta,v}(x) = \begin{pmatrix} \theta_{v+1} & \theta_{v+2} & \dots & \theta_{v+l} \\ \theta_{v+l+1} & \theta_{v+l+2} & \dots & \theta_{v+2l} \\ \vdots & \vdots & \vdots & \vdots \\ \theta_{v+(k-1)l+1} & \theta_{v+(k-1)l+2} & \dots & \theta_{v+kl} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_l \end{pmatrix} + \begin{pmatrix} \theta_{v+kl+1} \\ \theta_{v+kl+2} \\ \vdots \\ \theta_{v+kl+k} \end{pmatrix},$$

- $\forall\, \theta \in \mathbb{R}^\rho$ Funktion $\mathcal{U}_\theta\colon \mathbb{R}^d \to \mathbb{R}$ mit

$$\mathcal{U}_\theta = A_{1,d}^{\theta, sd(d+1)} \circ \mathcal{R} \circ A_{d,d}^{\theta,(s-1)d(d+1)} \circ \dots \circ \mathcal{R} \circ A_{d,d}^{\theta,d(d+1)} \circ \mathcal{R} \circ A_{d,d}^{\theta,0}.$$

Für $x \in \mathbb{R}^d$ und geeignete $\theta \in \mathbb{R}^\rho$ betrachten wir die Näherung $\mathcal{U}_\theta(x) \approx u(0, x) \in \mathbb{R}$.

Künstliches neuronales Netz mit $s \in \mathbb{N}$ verborgenen Schichten

- $\rho \in \mathbb{N}$ hinreichend groß,
- Funktion $\mathcal{R} \colon \mathbb{R}^d \to \mathbb{R}^d$ mit $\forall\, x = (x_1, \ldots, x_d) \in \mathbb{R}^d$:

$$\mathcal{R}(x) = \big(\max\{x_1, 0\}, \ldots, \max\{x_d, 0\}\big)\,,$$

- $\forall\, \theta = (\theta_1, \ldots, \theta_\rho) \in \mathbb{R}^\rho$, $v \in \mathbb{N}_0$, $k, l \in \mathbb{N}$ mit $v + k(l+1) \leq \rho$ Funktion $A^{\theta,v}_{k,l} \colon \mathbb{R}^l \to \mathbb{R}^k$ mit $\forall\, x = (x_1, \ldots, x_l) \in \mathbb{R}^l$:

$$A^{\theta,v}_{k,l}(x) = \begin{pmatrix} \theta_{v+1} & \theta_{v+2} & \cdots & \theta_{v+l} \\ \theta_{v+l+1} & \theta_{v+l+2} & \cdots & \theta_{v+2l} \\ \vdots & \vdots & \vdots & \vdots \\ \theta_{v+(k-1)l+1} & \theta_{v+(k-1)l+2} & \cdots & \theta_{v+kl} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_l \end{pmatrix} + \begin{pmatrix} \theta_{v+kl+1} \\ \theta_{v+kl+2} \\ \vdots \\ \theta_{v+kl+k} \end{pmatrix},$$

- $\forall\, \theta \in \mathbb{R}^\rho$ Funktion $\mathcal{U}_\theta \colon \mathbb{R}^d \to \mathbb{R}$ mit

$$\mathcal{U}_\theta = A^{\theta,sd(d+1)}_{1,d} \circ \mathcal{R} \circ A^{\theta,(s-1)d(d+1)}_{d,d} \circ \ldots \circ \mathcal{R} \circ A^{\theta,d(d+1)}_{d,d} \circ \mathcal{R} \circ A^{\theta,0}_{d,d}.$$

Für $x \in \mathbb{R}^d$ und geeignete $\theta \in \mathbb{R}^\rho$ betrachten wir die Näherung $\mathcal{U}_\theta(x) \approx u(0, x) \in \mathbb{R}$.

Künstliches neuronales Netz mit $s \in \mathbb{N}$ verborgenen Schichten

- $\rho \in \mathbb{N}$ hinreichend groß,
- Funktion $\mathcal{R}\colon \mathbb{R}^d \to \mathbb{R}^d$ mit $\forall\, x = (x_1, \ldots, x_d) \in \mathbb{R}^d$:

$$\mathcal{R}(x) = \left(\max\{x_1, 0\}, \ldots, \max\{x_d, 0\}\right),$$

- $\forall\, \theta = (\theta_1, \ldots, \theta_\rho) \in \mathbb{R}^\rho$, $v \in \mathbb{N}_0$, $k, l \in \mathbb{N}$ mit $v + k(l+1) \leq \rho$ Funktion $A^{\theta,v}_{k,l}\colon \mathbb{R}^l \to \mathbb{R}^k$ mit $\forall\, x = (x_1, \ldots, x_l) \in \mathbb{R}^l$:

$$A^{\theta,v}_{k,l}(x) = \begin{pmatrix} \theta_{v+1} & \theta_{v+2} & \cdots & \theta_{v+l} \\ \theta_{v+l+1} & \theta_{v+l+2} & \cdots & \theta_{v+2l} \\ \vdots & \vdots & \vdots & \vdots \\ \theta_{v+(k-1)l+1} & \theta_{v+(k-1)l+2} & \cdots & \theta_{v+kl} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_l \end{pmatrix} + \begin{pmatrix} \theta_{v+kl+1} \\ \theta_{v+kl+2} \\ \vdots \\ \theta_{v+kl+k} \end{pmatrix},$$

- $\forall\, \theta \in \mathbb{R}^\rho$ Funktion $\mathcal{U}_\theta\colon \mathbb{R}^d \to \mathbb{R}$ mit

$$\mathcal{U}_\theta = A^{\theta,sd(d+1)}_{1,d} \circ \mathcal{R} \circ A^{\theta,(s-1)d(d+1)}_{d,d} \circ \ldots \circ \mathcal{R} \circ A^{\theta,d(d+1)}_{d,d} \circ \mathcal{R} \circ A^{\theta,0}_{d,d}.$$

Für $x \in \mathbb{R}^d$ und geeignete $\theta \in \mathbb{R}^\rho$ betrachten wir die Näherung $\mathcal{U}_\theta(x) \approx u(0, x) \in \mathbb{R}$.

Künstliches neuronales Netz mit $s \in \mathbb{N}$ verborgenen Schichten

- $\rho \in \mathbb{N}$ hinreichend groß,
- Funktion $\mathcal{R}\colon \mathbb{R}^d \to \mathbb{R}^d$ mit $\forall\, x = (x_1, \ldots, x_d) \in \mathbb{R}^d$:

$$\mathcal{R}(x) = \big(\max\{x_1, 0\}, \ldots, \max\{x_d, 0\}\big),$$

- $\forall\, \theta = (\theta_1, \ldots, \theta_\rho) \in \mathbb{R}^\rho$, $v \in \mathbb{N}_0$, $k, l \in \mathbb{N}$ mit $v + k(l+1) \leq \rho$ Funktion $A^{\theta,v}_{k,l}\colon \mathbb{R}^l \to \mathbb{R}^k$ mit $\forall\, x = (x_1, \ldots, x_l) \in \mathbb{R}^l$:

$$A^{\theta,v}_{k,l}(x) = \begin{pmatrix} \theta_{v+1} & \theta_{v+2} & \cdots & \theta_{v+l} \\ \theta_{v+l+1} & \theta_{v+l+2} & \cdots & \theta_{v+2l} \\ \vdots & \vdots & \vdots & \vdots \\ \theta_{v+(k-1)l+1} & \theta_{v+(k-1)l+2} & \cdots & \theta_{v+kl} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_l \end{pmatrix} + \begin{pmatrix} \theta_{v+kl+1} \\ \theta_{v+kl+2} \\ \vdots \\ \theta_{v+kl+k} \end{pmatrix},$$

- $\forall\, \theta \in \mathbb{R}^\rho$ Funktion $\mathcal{U}_\theta\colon \mathbb{R}^d \to \mathbb{R}$ mit

$$\mathcal{U}_\theta = A^{\theta,sd(d+1)}_{1,d} \circ \mathcal{R} \circ A^{\theta,(s-1)d(d+1)}_{d,d} \circ \ldots \circ \mathcal{R} \circ A^{\theta,d(d+1)}_{d,d} \circ \mathcal{R} \circ A^{\theta,0}_{d,d}.$$

Für $x \in \mathbb{R}^d$ und geeignete $\theta \in \mathbb{R}^\rho$ betrachten wir die Näherung $\mathcal{U}_\theta(x) \approx u(0,x) \in \mathbb{R}$.

Künstliches neuronales Netz mit $s \in \mathbb{N}$ verborgenen Schichten

- $\rho \in \mathbb{N}$ hinreichend groß,
- Funktion $\mathcal{R} \colon \mathbb{R}^d \to \mathbb{R}^d$ mit $\forall\, x = (x_1, \dots, x_d) \in \mathbb{R}^d$:

$$\mathcal{R}(x) = \left(\max\{x_1, 0\}, \dots, \max\{x_d, 0\}\right),$$

- $\forall\, \theta = (\theta_1, \dots, \theta_\rho) \in \mathbb{R}^\rho$, $v \in \mathbb{N}_0$, $k, l \in \mathbb{N}$ mit $v + k(l+1) \le \rho$ Funktion $A^{\theta,v}_{k,l} \colon \mathbb{R}^l \to \mathbb{R}^k$ mit $\forall\, x = (x_1, \dots, x_l) \in \mathbb{R}^l$:

$$A^{\theta,v}_{k,l}(x) = \begin{pmatrix} \theta_{v+1} & \theta_{v+2} & \dots & \theta_{v+l} \\ \theta_{v+l+1} & \theta_{v+l+2} & \dots & \theta_{v+2l} \\ \vdots & \vdots & \vdots & \vdots \\ \theta_{v+(k-1)l+1} & \theta_{v+(k-1)l+2} & \dots & \theta_{v+kl} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_l \end{pmatrix} + \begin{pmatrix} \theta_{v+kl+1} \\ \theta_{v+kl+2} \\ \vdots \\ \theta_{v+kl+k} \end{pmatrix},$$

- $\forall\, \theta \in \mathbb{R}^\rho$ Funktion $\mathcal{U}_\theta \colon \mathbb{R}^d \to \mathbb{R}$ mit

$$\mathcal{U}_\theta = A^{\theta,sd(d+1)}_{1,d} \circ \mathcal{R} \circ A^{\theta,(s-1)d(d+1)}_{d,d} \circ \dots \circ \mathcal{R} \circ A^{\theta,d(d+1)}_{d,d} \circ \mathcal{R} \circ A^{\theta,0}_{d,d}.$$

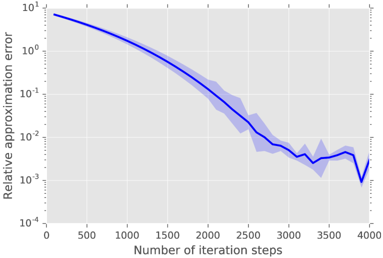
Für $x \in \mathbb{R}^d$ und geeignete $\theta \in \mathbb{R}^\rho$ betrachten wir die Näherung $\mathcal{U}_\theta(x) \approx u(0,x) \in \mathbb{R}$.

Sowohl numerische Simulationen als auch erste theoretischen Ergebnisse legen die Vermutung nahe, dass **künstliche neuronale Netze** in vielen Fällen den Fluch der Dimensionalität überwinden.

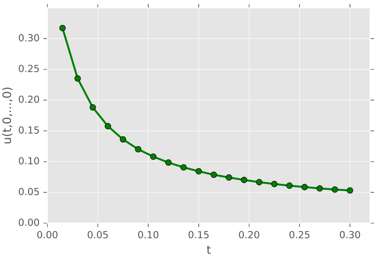
100-dimensionale Allen-Cahn Gleichung

$$\frac{\partial u}{\partial t}(t,x) = (\Delta_x u)(t,x) + u(t,x) - [u(t,x)]^3$$

mit $u(0,x) = \frac{1}{(2+0.4\|x\|^2)}$ für $t \in [0, \frac{3}{10}]$, $x \in \mathbb{R}^{100}$.



(a) Relativer Fehler für $u(\frac{3}{10}, 0) \approx 0.0528$



(b) Näherungsw. Darst. von $u(t,0)$, $0 \leq t \leq \frac{3}{10}$

Fehler: 0.3%, **Laufzeit:** 647 **Sekunden**. PYTHON, TENSORFLOW, MACBOOK PRO 2.9 GHz (INTEL i5, 16 GB RAM)

Details finden sich in

- C. Beck, W. E, & A. Jentzen. Machine learning approximation algorithms for high-dimensional fully nonlinear partial differential equations and second-order backward stochastic differential equations. *arXiv* (2017).
- R. E. Bellman. Dynamic programming. *Princeton University Press* (1957).
- W. E, J. Han, & A. Jentzen. Deep learning-based numerical methods for high-dimensional parabolic partial differential equations and backward stochastic differential equations. *Comm. Math. Stat.* (2017).
- J. Han, A. Jentzen, & W. E. Overcoming the curse of dimensionality: Solving high-dimensional partial differential equations using deep learning. *arXiv* (2017).
- D. J. Higham. An introduction to financial option valuation. *Cambridge University Press* (2004).
- J. Kallsen. Option pricing. In T. Andersen, R. Davis, J. Kreiß, and T. Mikosch, Handbook of Financial Time Series, *Springer* (2009).

und in den dort genannten Referenzen.

Vielen Dank für Ihre Aufmerksamkeit!

Details finden sich in

- C. Beck, W. E, & A. Jentzen. Machine learning approximation algorithms for high-dimensional fully nonlinear partial differential equations and second-order backward stochastic differential equations. *arXiv* (2017).
- R. E. Bellman. Dynamic programming. *Princeton University Press* (1957).
- W. E, J. Han, & A. Jentzen. Deep learning-based numerical methods for high-dimensional parabolic partial differential equations and backward stochastic differential equations. *Comm. Math. Stat.* (2017).
- J. Han, A. Jentzen, & W. E. Overcoming the curse of dimensionality: Solving high-dimensional partial differential equations using deep learning. *arXiv* (2017).
- D. J. Higham. An introduction to financial option valuation. *Cambridge University Press* (2004).
- J. Kallsen. Option pricing. In T. Andersen, R. Davis, J. Kreiß, and T. Mikosch, Handbook of Financial Time Series, *Springer* (2009).

und in den dort genannten Referenzen.

Vielen Dank für Ihre Aufmerksamkeit!