



Neuronale Netze – Kalibrierung von Kapitalmarktmodellen

Mascha Fiona Baedorf und Marco Getrost

23.07.2019



Inhalt

I	Neuronale Netze
II	ESG Kalibrierung mit neuronalen Netzen

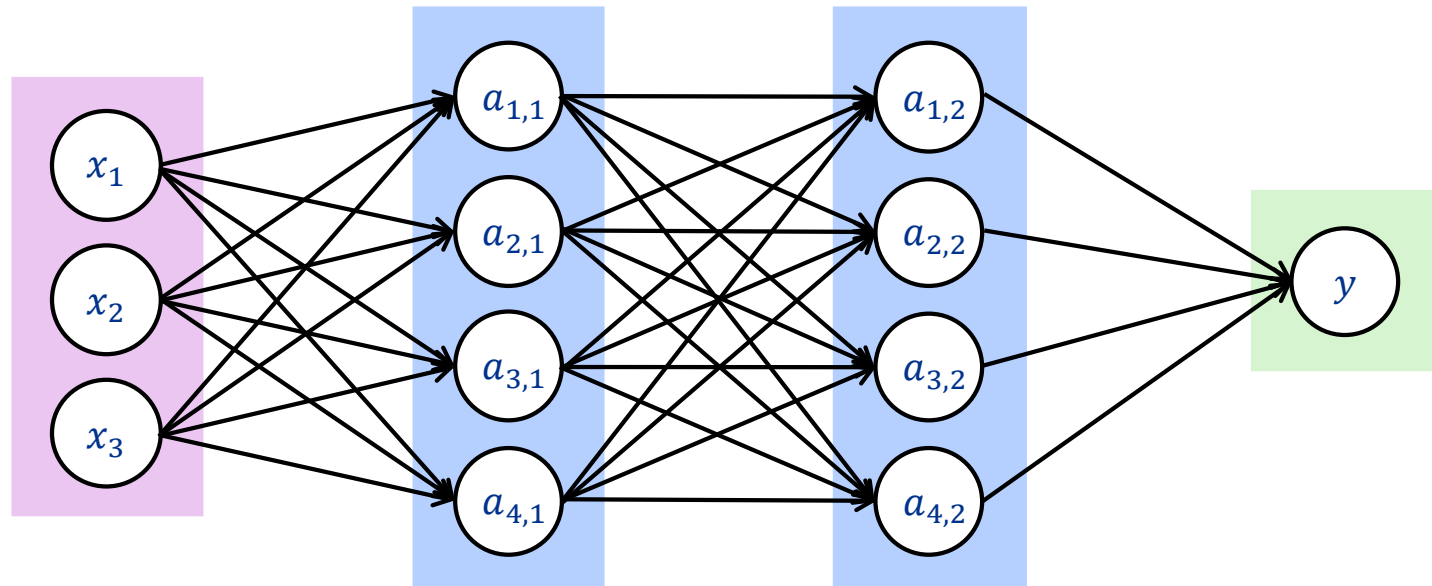
Künstliche Neuronale Netze

Input-Layer

Hidden-Layer

Hidden-Layer

Output-Layer



$$a_{i,1} = \sum_{j=1}^3 x_j \cdot w_{j,i}$$

$$a_{i,2} = \sum_{j=1}^4 a_{j,1} \cdot w_{j,i}$$

$$y = \sum_{j=1}^4 a_{j,2} \cdot w_{j,i}$$

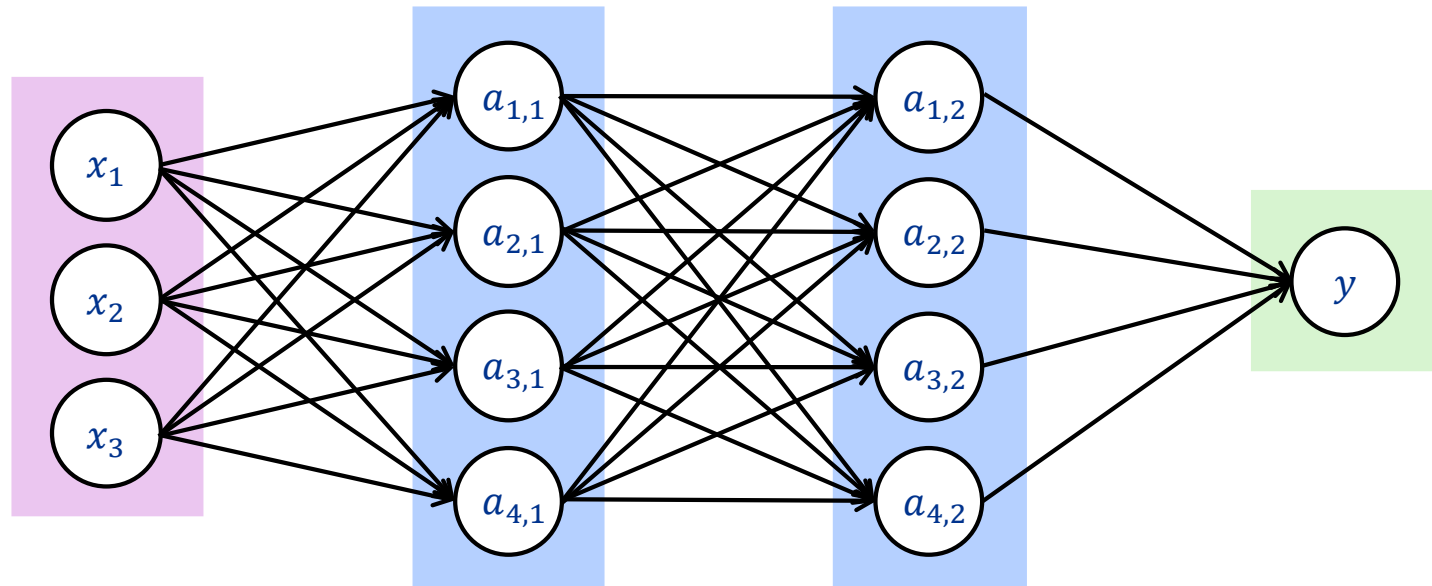
Künstliche Neuronale Netze

Input-Layer

Hidden-Layer

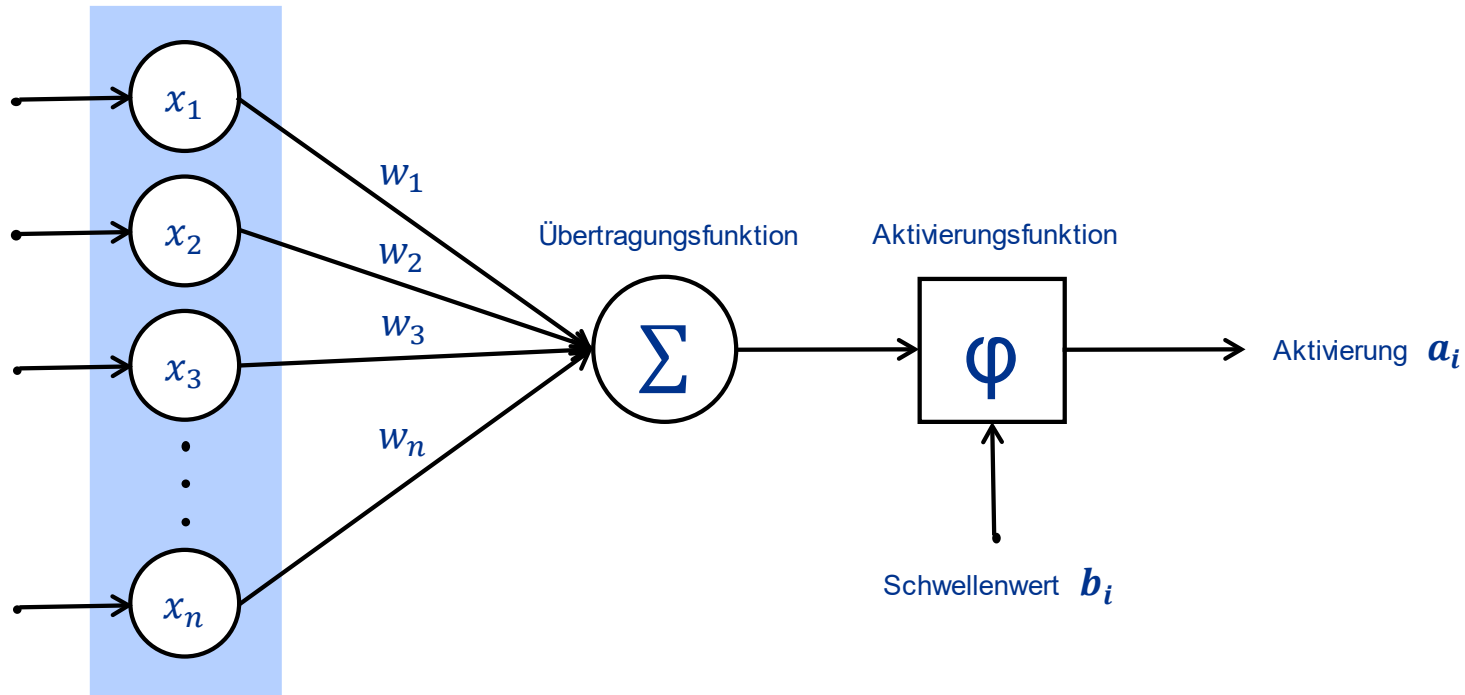
Hidden-Layer

Output-Layer



$$a_{i,1} = \varphi \left(\sum_{j=1}^3 x_j \cdot w_{j,i} \right) \quad a_{i,2} = \varphi \left(\sum_{j=1}^4 a_{j,1} \cdot w_{j,i} \right) \quad y = \varphi \left(\sum_{j=1}^4 a_{j,2} \cdot w_{j,1} \right)$$

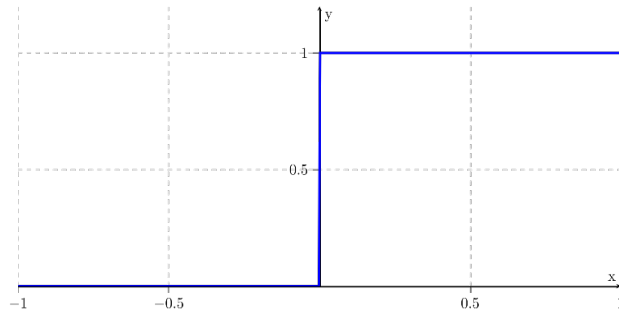
Aufbau der Neuronen



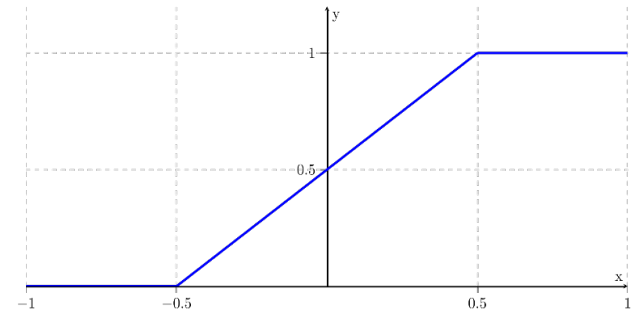
$$a_i = \varphi \left(\sum_{j=1}^n w_j \cdot x_j - b_i \right)$$

Aktivierungsfunktionen

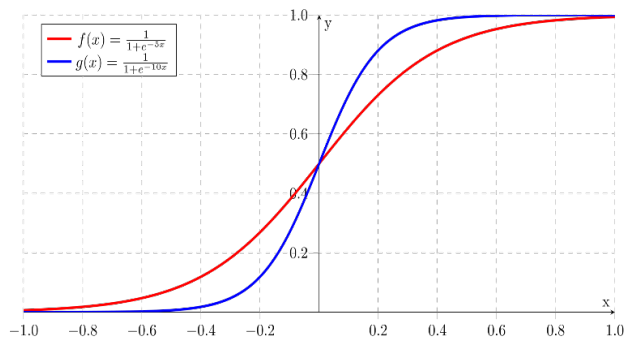
Schwellenwertfunktion



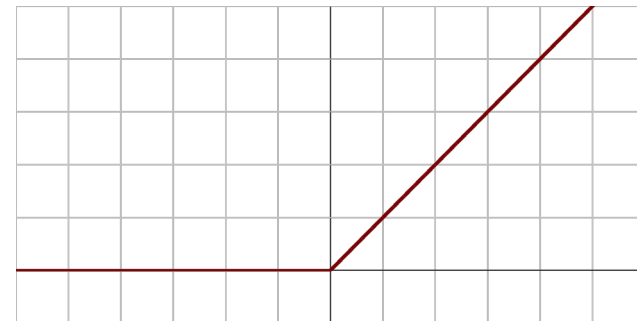
Stückweise lineare Funktion



Sigmoidfunktion



Rectifier (ReLU)



Verlustfunktionen

L1-Fehler:

$$\mathcal{L} = \sum_{i=1}^n |y_i - t_i|$$

L2-Fehler:

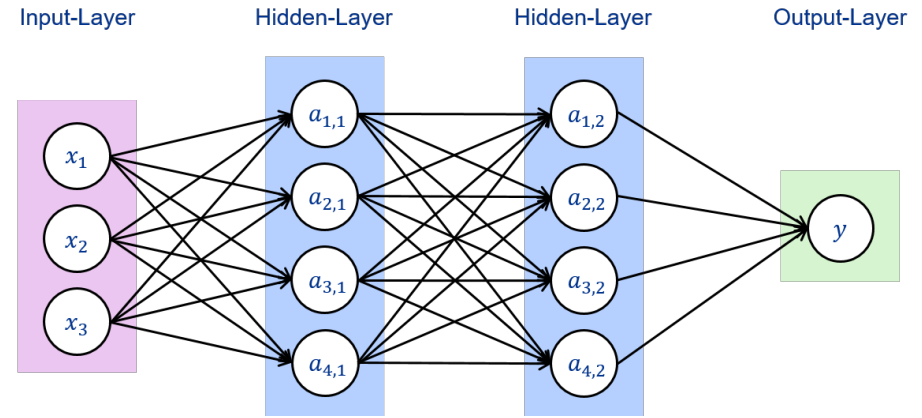
$$\mathcal{L} = \sum_{i=1}^n (y_i - t_i)^2$$

Hinge (Klassifizierung, SVM):

$$\mathcal{L} = \sum_{i=1}^n \max(0, 1 - t_i \cdot y_i) \quad \text{mit } t_i = \pm 1$$

Kullback Leibler (Verteilung):

$$\mathcal{L} = \sum_{i=1}^n (t_i \cdot \log t_i) - \sum_{i=1}^n (t_i \cdot \log y_i)$$



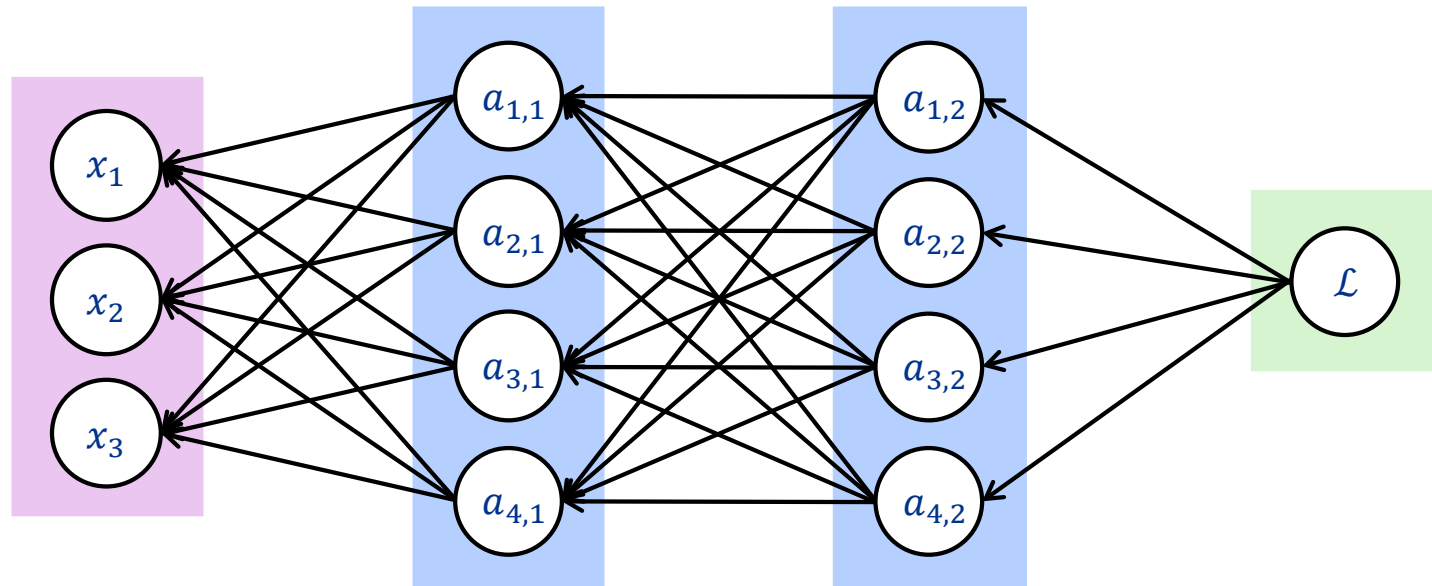
Backpropagation – Das Netz lernt

Input-Layer

Hidden-Layer

Hidden-Layer

Output-Layer



Fehleranteil:

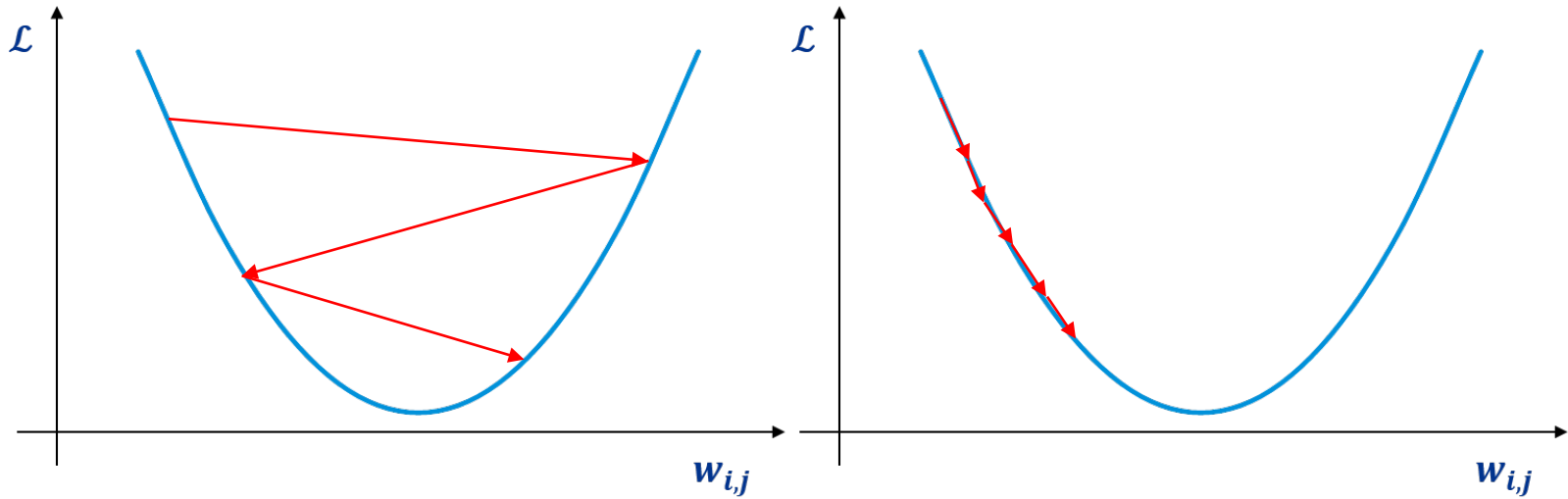
$$\Delta w_{i,j} = -\lambda \cdot \frac{\delta \mathcal{L}}{\delta w_{i,j}}$$

Mit $\lambda \triangleq$ Lernrate

Update:

$$w_{i,j}^{neu} = w_{i,j}^{alt} + \Delta w_{i,j}$$

Lernrate λ

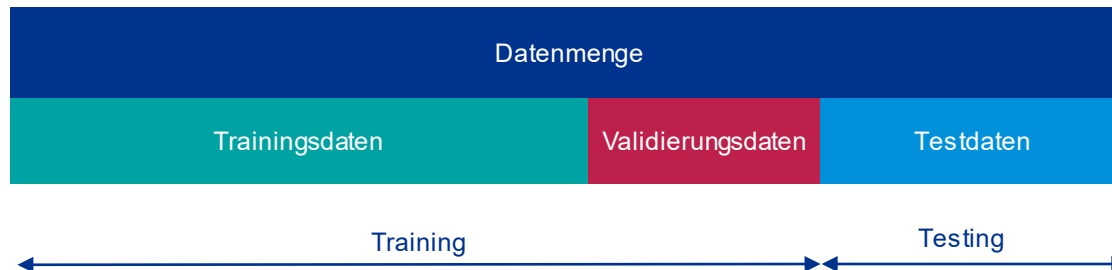
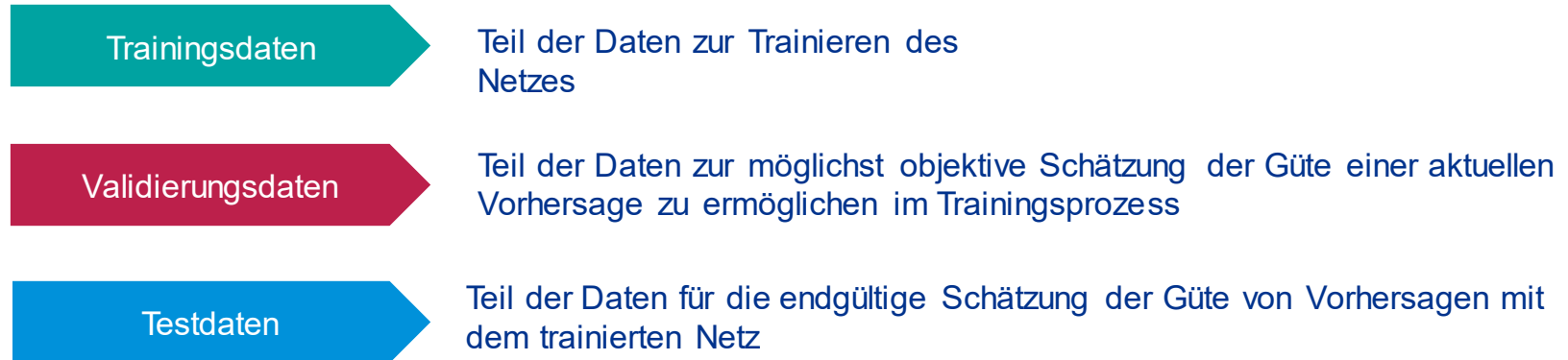


Zu große Lernrate kann zu „overshooting“ führen

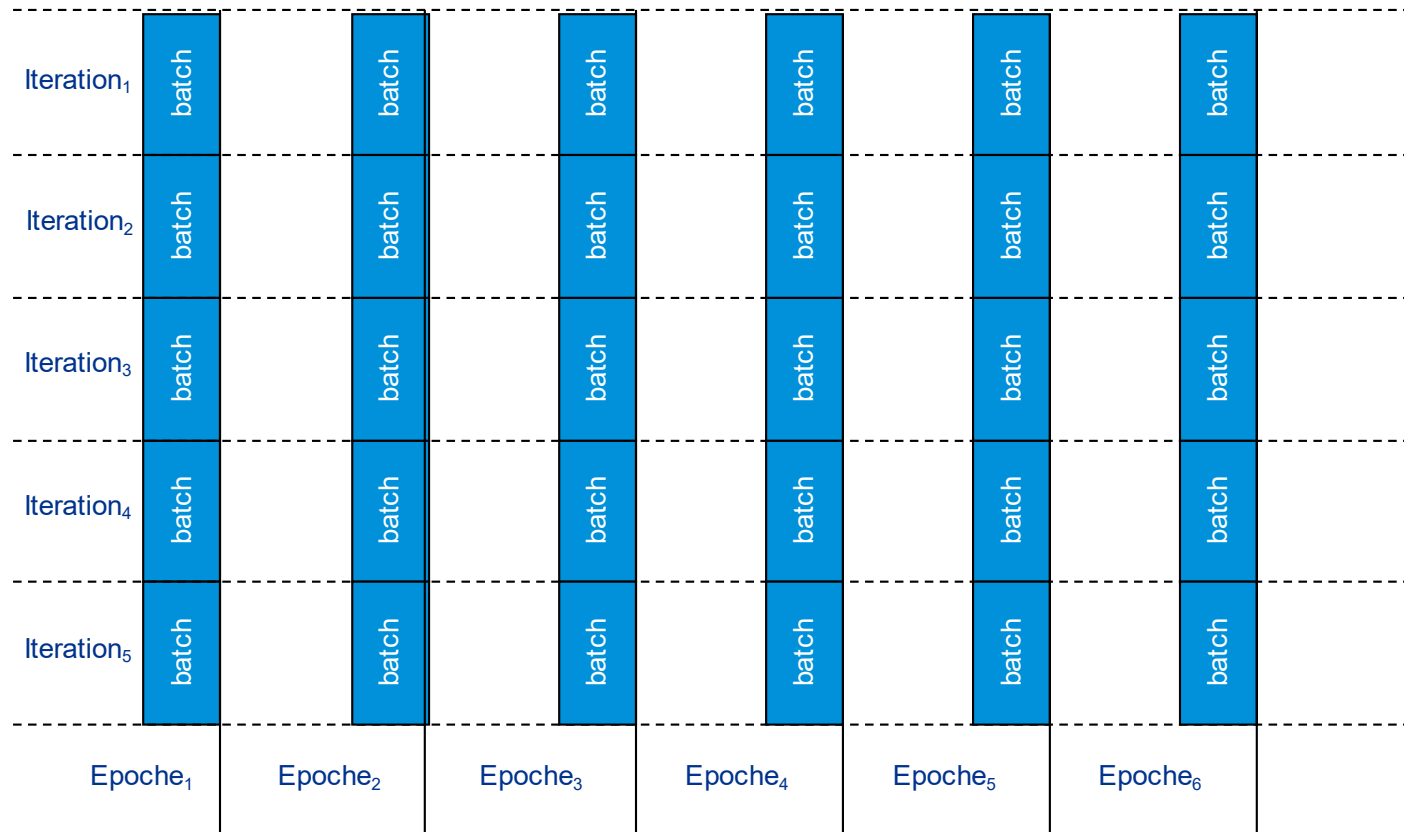
Zu kleine Lernrate kann zu einer langsamen Konvergenz führen.

$$w_{i,j}^{neu} = w_{i,j}^{alt} - \boxed{\lambda} \cdot \frac{\delta \mathcal{L}}{\delta w_{i,j}}$$

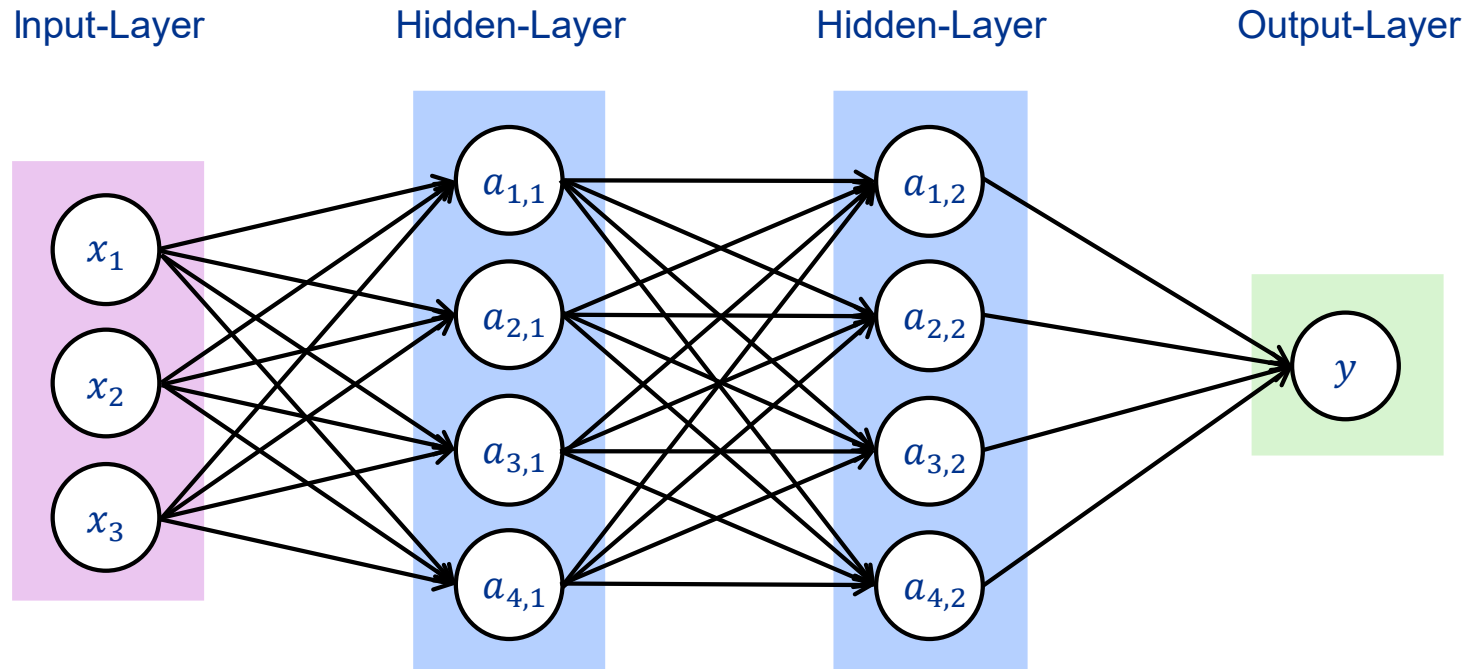
Trainings-, Validierungs-, Testdaten



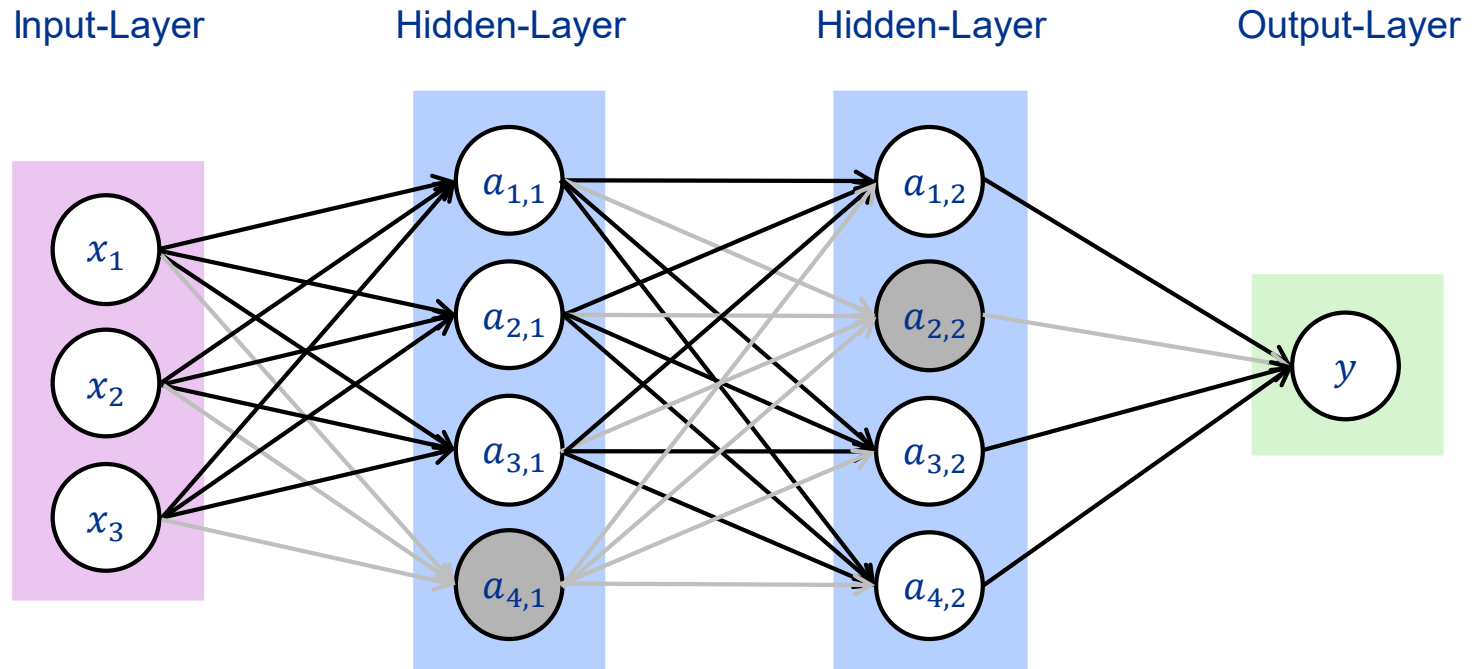
Epochen, Iteration, Batch



Drop-out-rate

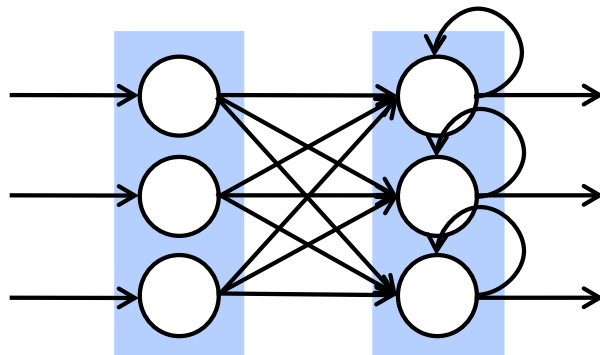


Drop-out-rate

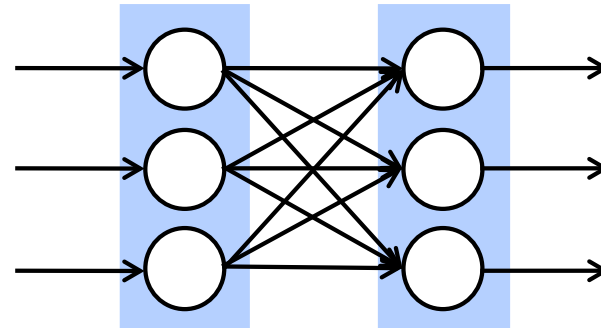


Rekurrentes Netz

- Sind für die Arbeit mit **sequentiellen Daten** angepasst, d.h. Daten, bei denen verwandte Dinge aufeinander folgen
- Wichtigster Unterschied zu Feed-Forward-Netzen: die Informationen laufen hier durch **eine Schleife** → versteckte Schichten können sich nicht nur an die aktuelle Ausgabe, sondern auch an **die Ausgabe einen Schritt vorher erinnern**
- Weiterer wichtiger Unterschied: Gewichtung kann durch die Verwendung des **Backpropagationsschemas** angepasst werden



Recurrent Neural Network



Feed-Forward Neural Network

Inhalt

I	Neuronale Netze
II	ESG Kalibrierung mit neuronalen Netzen

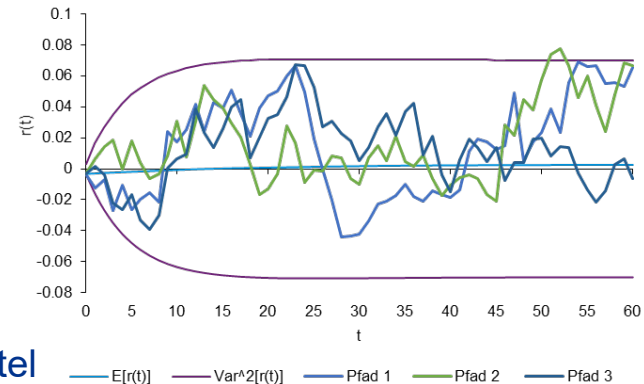
Zinsmodell: Hull White 1-Faktor Modell

Modellbeschreibung

- Short Rate entspricht Zins für infinitesimal kleinen Zeitraum
- Differentialgleichung: $dr(t) = [\theta(t) - \alpha r(t)]dt + \sigma dW_t$
- $r(t) \sim N\left(e^{-\alpha t} r(0) + \int_0^t e^{-\alpha(s-t)} \theta(s) ds, \frac{\sigma}{2\alpha} (1 - e^{-2\alpha t})\right)$

Vorteile

- Einfache Kalibrierung
- Möglichkeit, negative Zinsen zu modellieren
- geschlossene Formeln für einfache Optionen auf Zinstitel



Nachteile

- Hohe Anzahl negativer Zinsen
- Vollständige Korrelation der einzelnen Laufzeiten
- Wenig Dynamik in der Zinskurve
- Eingeschränkte Anzahl an Freiheitsgraden (α, σ) zur Kalibrierung an Swaptions
- Häufig nur Kalibrierung mit σ und $\alpha = 0.1$

Kalibrierung von Zinsmodellen

Für die Kalibrierung

1. Verwendung von finanzmathematische Zusammenhänge mit numerischer Optimierung

2. Verwendung von neuronalen Netzen

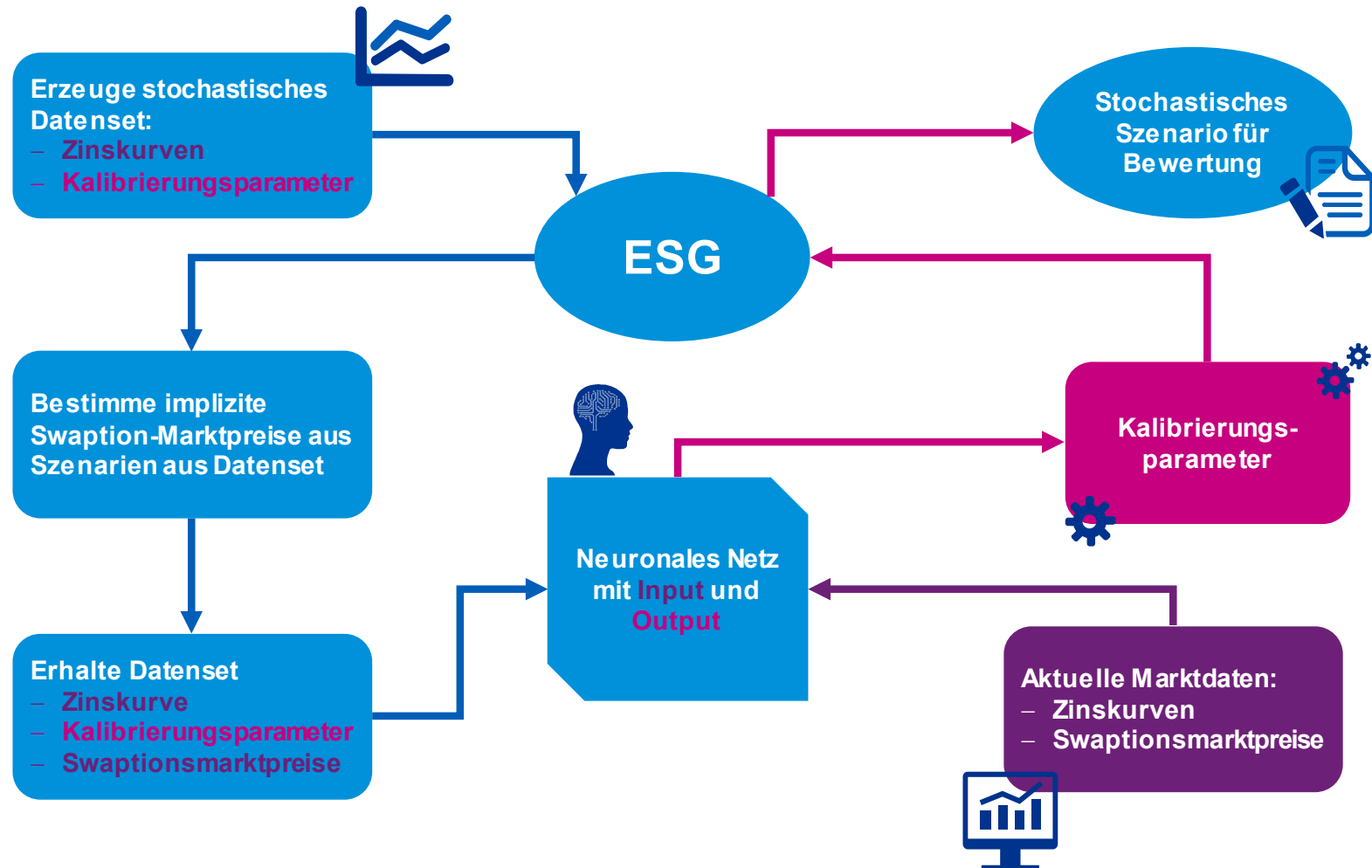
Idee: Trainiere neuronales Netz mit:

- Input: Marktparameter (Zinskurve, Marktpreise bzw. Volatilitäten)
- Output: Freiheitsgerade des Zinsmodells (Hull White 1-Faktor: σ)

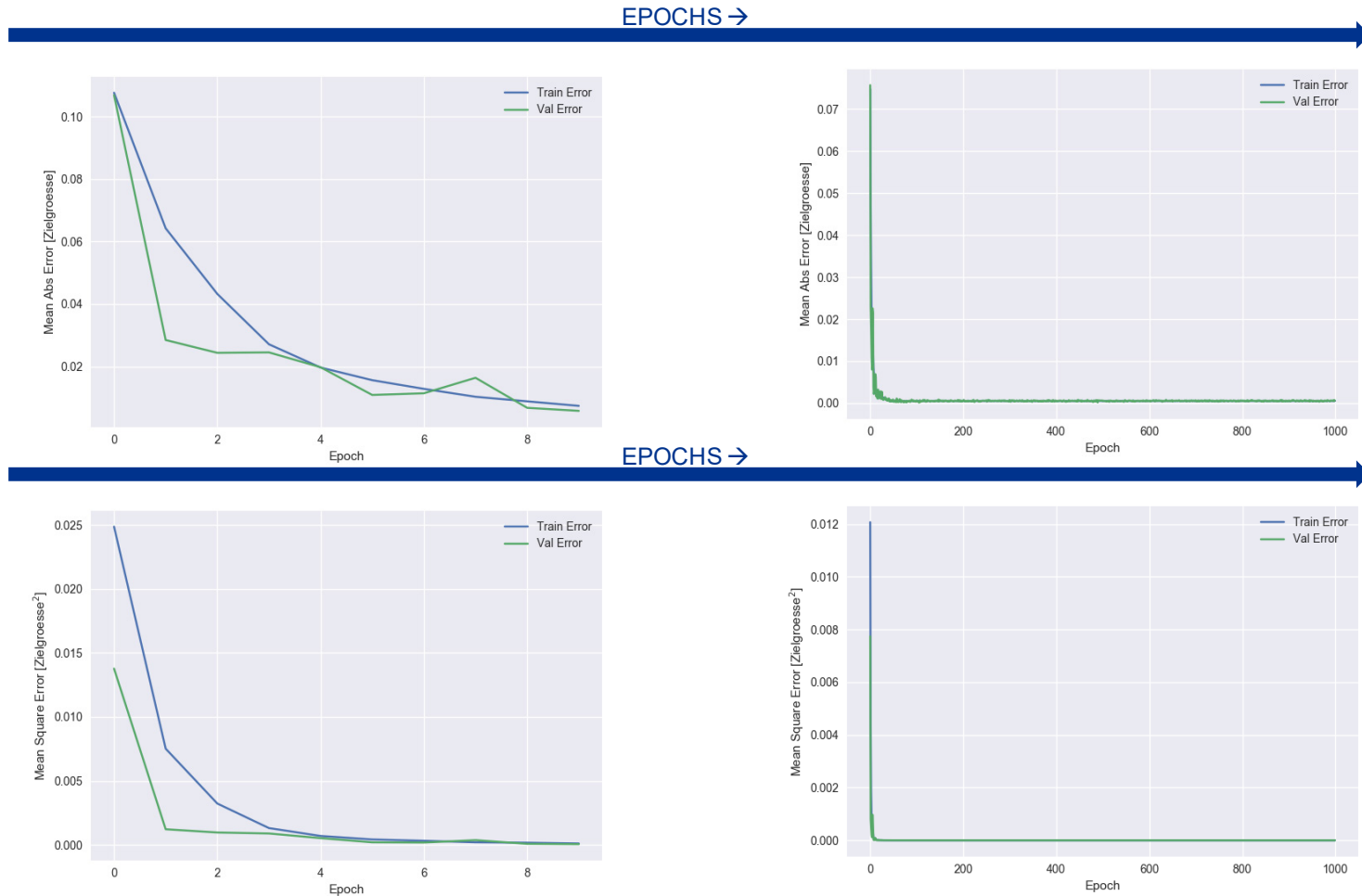
Vorgehen:

- Erzeuge Szenarien unter Variation der Zinskurve und den Freiheitsgraden
- Berechne die aus den Szenarien resultierende Größen (Swaptionspreise)
→ Datenset mit Input und Output für neuronales Netz

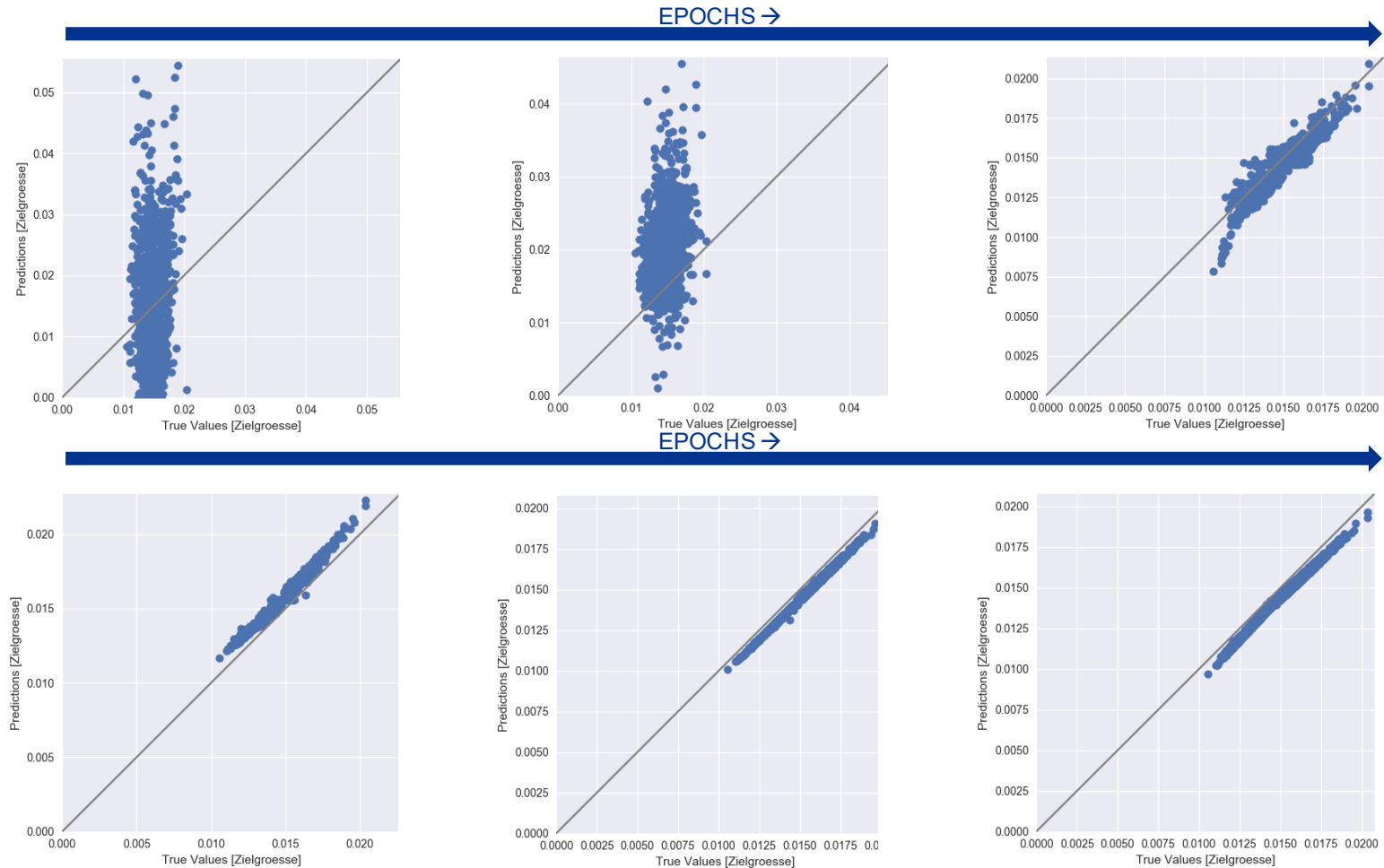
Kalibrierung von Zinsmodellen



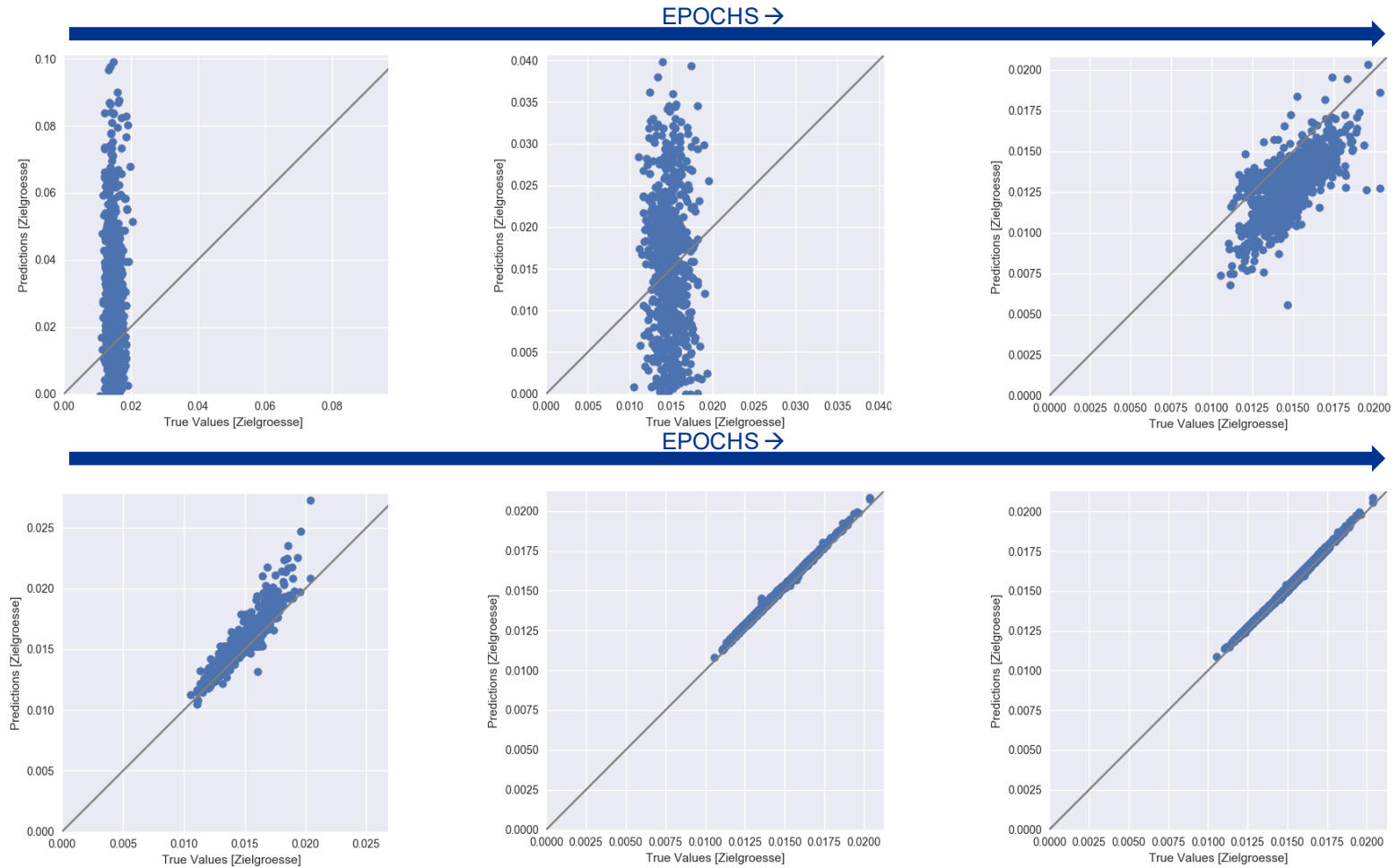
Ergebnisse (Training/Validation Error)



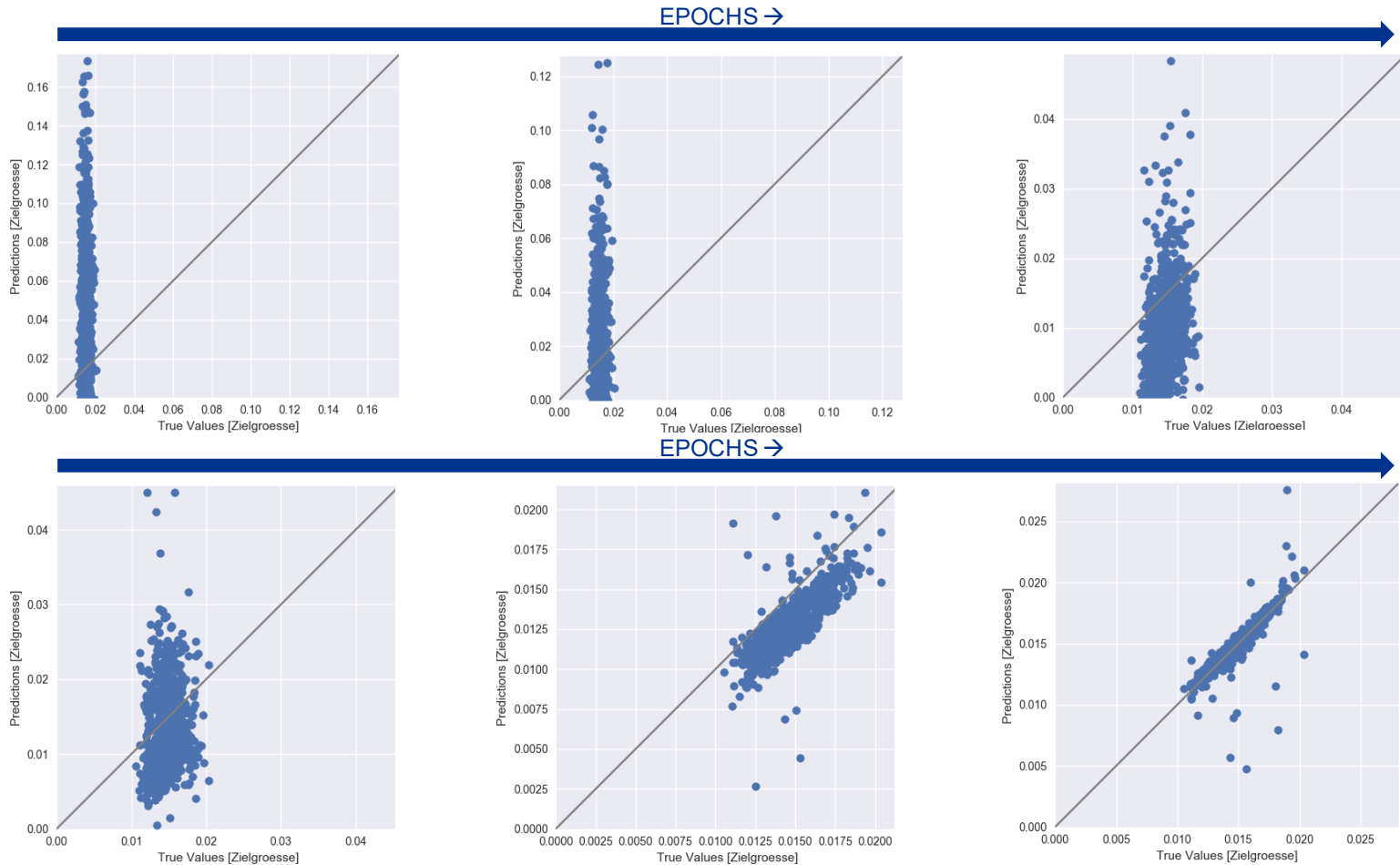
Ergebnisse (Test Error) Learn Rate = 0.001



Ergebnisse (Test Error) Learn Rate = 0.0005

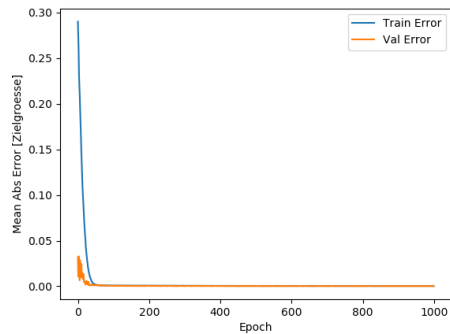


Ergebnisse (Test Error) Learn Rate = 0.0001

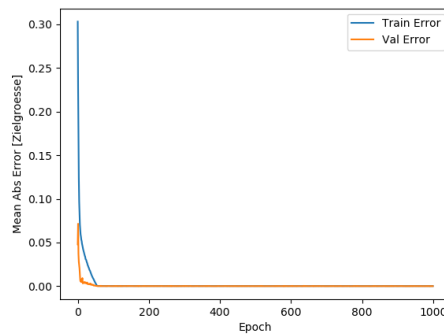


Ergebnisse (Test Error) Learn Rate = 0.0001

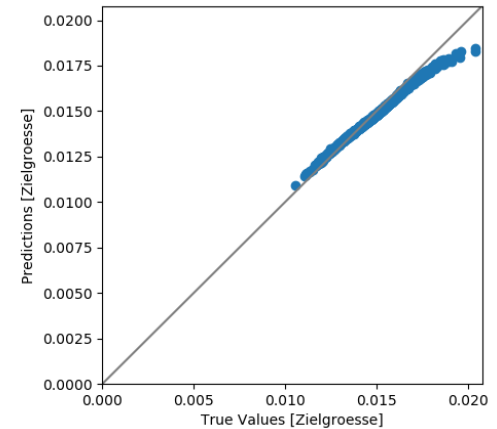
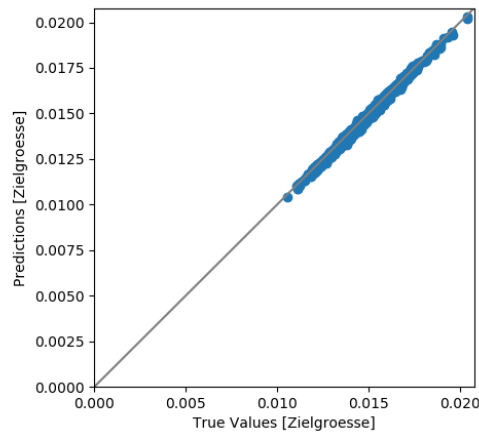
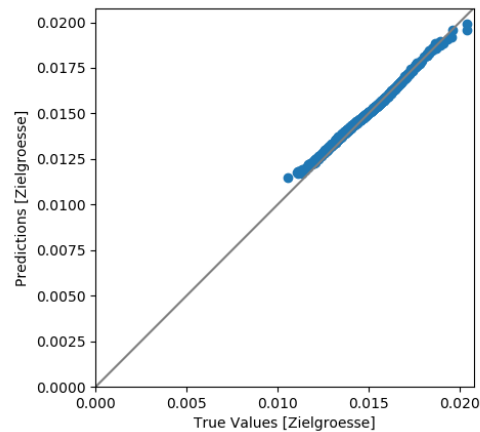
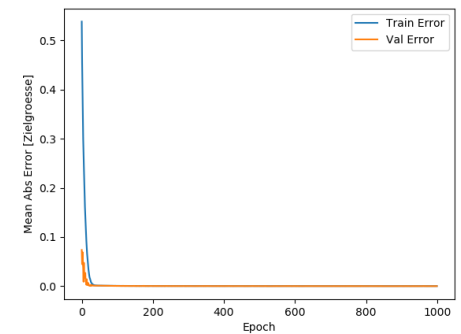
activation = Sigmoid



activation = tanh



activation = softplus



Ansprechpartner

Mascha Fiona Baedorf

Assistant Manager, Financial Services
T +49 151 22342773
mbaedorf@kpmg.com

KPMG AG Wirtschaftsprüfungsgesellschaft
Barbarossaplatz 1a
50674 Köln

Marco Getrost

Senior Associate, Financial Services
T +49 160 3101028
mgetrost@kpmg.com

KPMG AG Wirtschaftsprüfungsgesellschaft
Barbarossaplatz 1a
50674 Köln



www.kpmg.de/socialmedia

www.kpmg.de

Die enthaltenen Informationen sind allgemeiner Natur und nicht auf die spezielle Situation einer Einzelperson oder einer juristischen Person ausgerichtet. Obwohl wir uns bemühen, zuverlässige und aktuelle Informationen zu liefern, können wir nicht garantieren, dass diese Informationen so zutreffend sind wie zum Zeitpunkt ihres Eingangs oder dass sie auch in Zukunft so zutreffend sein werden. Niemand sollte aufgrund dieser Informationen handeln ohne geeigneten fachlichen Rat und ohne gründliche Analyse der betreffenden Situation.

© 2016 KPMG AG Wirtschaftsprüfungsgesellschaft, ein Mitglied des KPMG-Netzwerks unabhängiger Mitgliedsfirmen, die KPMG International Cooperative („KPMG International“), einer juristischen Person schweizerischen Rechts, angeschlossen sind. Alle Rechte vorbehalten. Der Name KPMG und das Logo sind eingetragene Markenzeichen von KPMG International.