



# PYTPRA

# Maschinelles Lernen in der Praxis

# Logit

# Logistische Regression

- Wir wollen dazu ein Laborexperiment mit dem Jupyter Notebook **PYTPRA\_03\_01\_LogitExperiment.ipynb** durchführen
- Unterstellt wird, dass wir  $n$  Subjekte haben, die wir beobachten
- In diesen Wesen baut sich etwa eine Erregung „**Agitation**“ **a** auf, die wir nur mittelbar beobachten können. Die **Variable** **a** ist also **latent**.
- Überschreitet sie ein bestimmtes Niveau, z B die Zahl 0, so fällt das Subjekt eine Entscheidung, z B **Kauf** eines Kfz = **buy**. Diese Entscheidung können wir beobachten.
- Die latente Variable kann je nach Situation anders aussehen.
- Bei den Bankkunden könnte sie die Bonität darstellen. Steigt die Bonität über ein bestimmtes Niveau, so wird aus einem Sub-Prime Kunden ein Prime Kunde (oder von Sub-Investment Grade zu IG). Dabei wird die Bonität durch verschiedene Faktoren wie Einkommen, Alter usw. beeinflusst
- In der Medizin könnte die latente Variable „Bio-Stress“ sein – beeinflusst durch die Lebenssituation – und das Beobachtbare ob ein Schlaganfall auftritt oder nicht

# Logistische Regression

Unterstelle also eine latente Variable  $a_i$  (Agitation = Erregung) für Subjekt  $i$  wie folgt

$$a_i = c + b x_i + \varepsilon_i$$

Der Störterm ist logistisch verteilt. Dies hat Praktikabilitätsgründe.

Die ZV  $a_i$  bestimmt das beobachtbare binäre  $y_i$ :  $y_i = 1$  wenn  $a_i > 0$ , 0 sonst

Die Wahrscheinlichkeit dafür ist:

$$P\{ y_i^* > 0 \mid x_i \} = P\{ -\varepsilon_i \leq c + b x_i \mid x_i \} = F(..),$$

$F$  ist die Verteilungsfunktion für  $-\varepsilon_i$ , also die logistische Verteilung.

Der Graph der logistischen Dichte ist ähnlich wie die Gauss-Glocke – nur etwas spitzer und höher.

# Logistische Regression

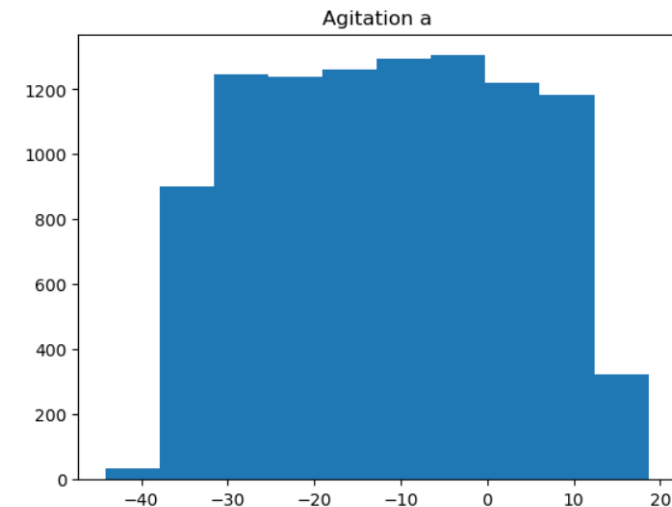
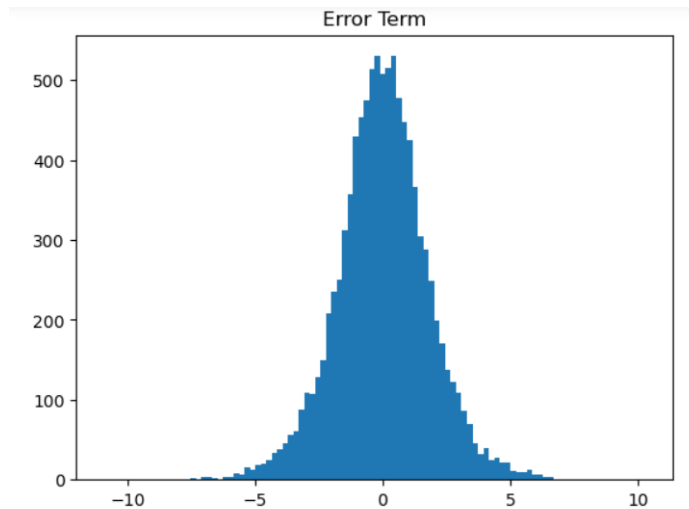
Unterstelle also eine latente Variable  $a_i$  (Agitation = Erregung) für Subjekt  $i$  wie folgt

$$a_i = c + b x_i + \varepsilon_i$$

$y_i = 1$  wenn  $a_i > 0$ , 0 sonst

**Was werden wir am häufigsten beobachten? 0 o. 1?**

```
# set up experimental data
n=10000 #size of training sample
np.random.seed(seed=2024)
eps = np.random.logistic(loc=0.0, scale=1.0, size=(n,1))
#pi/sqrt(3)) is another scale setting usual in logit models
x = np.random.randint(18, 80, n)
x = x.reshape(x.size,1)
#binary derived from latent
c=-50
b=0.8
#latent variable with logistic error
a=c+b*x+eps
```

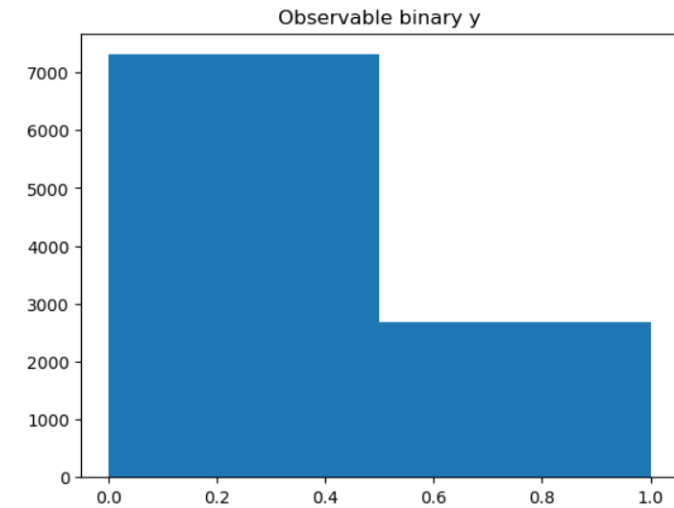


# Logistische Regression

Basierend auf den Datensätzen  $(y, x)$  gibt es ein Schätzverfahren (MaxLik), um die sog. Logistische Regression auszuführen

Logit Regression Results

=====						
Dep. Variable:		y	No. Observations:		10000	
Model:		Logit	Df Residuals:		9998	
Method:		MLE	Df Model:		1	
Date:		Thu, 03 Aug 2023	Pseudo R-squ.:		0.8935	
Time:		16:22:53	Log-Likelihood:		-620.54	
converged:		True	LL-Null:		-5824.6	
Covariance Type:		nonrobust	LLR p-value:		0.000	
=====						
	coef	std err	z	P> z	[0.025	0.975]
-----						
const	-51.3918	2.036	-25.237	0.000	-55.383	-47.401
x1	0.8217	0.033	25.230	0.000	0.758	0.886
=====						



The LLR p-value provides the p-value from a likelihood-ratio test of the model at hand versus the null model  
 $H_0: \text{all } \beta = 0$

The p-value correspond to a Wald test  
 $H_0: \alpha = 0 \text{ and } \beta_k = 0$

**Wurde der datengenerierende Prozess  $y(x)$  erkannt?**  
**Ändern Sie den Seed auf Ihr Geburtsjahr!**

# Theorem Maximum Likelihood Schätzung, Big Data Version

Wenn das Modell richtig spezifiziert ist, d.h.

- die richtige „Linkfunktion“  $F$  (und damit implizit das korrekte latente Modell)
- keine Autokorrelation und/oder Heteroskedastizität in den Störtermen
- keine Abhängigkeit zwischen Störterm und Regressoren
- Keine wichtigen Regressoren vergessen

Dann sind die Maximum - Likelihood - Schätzer

1. Konsistent (Definition wie bei lin. Regressionsmodell, d.h. schwache Konsistenz, mit wachsendem Stichprobenumfang konvergiert der Schätzer stochastisch gg. Wahren Wert = WS für Abweichungen vom wahren Wert geht gegen Null)
2. Asymptotisch effizient (kleinste Varianz unter allen konsistenten)
3. Asymptotisch normal verteilt

Bemerkung: Gültigkeit von 1+3 gut für Wald Test (Verbeek, 2012,. 189)

	coef	std err	z	P> z
const	-51.3918	2.036	-25.237	0.000
x1	0.8217	0.033	25.230	0.000

# Modellgüte

- Basis ist die **Konfusionsmatrix**
- Wir tragen in der Vertikalen die Vorhersagen ab (yhat)

		Realität: 0    1	
Modellprognose	0	<b>Confusion Matrix :</b> [[7174   134] [ 142 2550]]	
	1		

- Addiert man entlang der Hauptdiagonalen ergibt sich 9724, d.h. 97,24% Genauigkeit (Accuracy) der Fälle wurden in-sample korrekt vorhergesagt
- Diese Zahl ist gegen die naïve Prognose des häufigsten Falls zu benchmarken
- Da 73,08% der  $y_i$  eine 0 sind, führt die stete Vorhersage einer 0 zu einer Genauigkeit von 73,08%
- **Beweisen Sie dies mit der Matrix!**

# Modellgüte

- Je nach Situation sind andere Maßzahlen als die Genauigkeit wichtiger, die auf FN etc basieren

		True Class	
		Positive	Negative
Predicted Class	Positive	True Positive (TP)	False Positive (FP)
	Negative	False Negative (FN)	True Negative (TN)

- Man beachte **stets** die **Beschriftung** der Zeilen und Spalten und deren Reihenfolge
- In R beginnt die Konfusionsmatrix mit der Negative-Zeile (Y= 0) und dann folgt darunter die Positive-Zeile



# Modellgüte

- Je nach Situation sind andere Maßzahlen als die Genauigkeit wichtiger, z B bzgl Kreditausfall bei Bankkunden

Confusion matrix (also named contingency table or error matrix) displays predictive performance

		REALITY		
		True	False	
Fore cast T  F	T	True Positive (TP)	False Positive (FP) → Type I Error → False Alarm	Precision or Positive Predictive Value $= \frac{TP}{TP+FP}$
	F	False Negative (FN) → Type II Error / Miss	True Negative (TN)	
		Sensitivity <sup>†</sup> = TP Rate $= \frac{TP}{TP+FN}$	Specificity = TN Rate $= \frac{TN}{FP+TN}$	Accuracy $= \frac{TP+TN}{\text{Total}}$

<sup>†</sup> Equivalent with hit rate and recall

Data Mining, with Linear Discriminants: Prediction Performance

35

Example: Blood probe to test for cancer

		Patient with Cancer		
		True	False	
Positive Blood Test Outcome	True	TP: Cancer correctly diagnosed	FP: Healthy person diagnosed cancer	Precision $= \frac{TP}{TP+FP}$
	False			
Negative Blood Test Outcome	True	FN: Cancer not diagnosed	TN: Healthy person diagnosed as healthy	
	False			
		Sensitivity = TP Rate $= \frac{TP}{TP+FN}$	Specificity = TN Rate $= \frac{TN}{FP+TN}$	Accuracy $= \frac{TP+TN}{\text{Total}}$

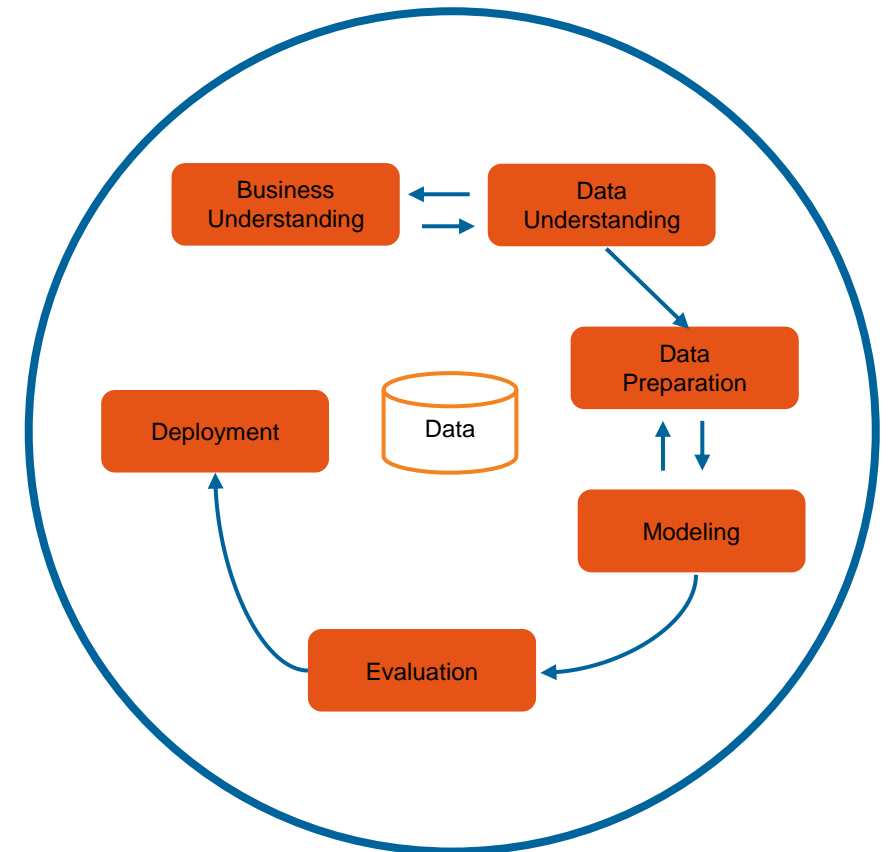
Different loss functions: Redundant, € 1000 check in FP case, compared to lethal outcome in FN case

# Cross Industry Standard Process for Data Mining

- Wir wollen auf Echtdaten arbeiten mit dem Notebook **PYTPRA\_03\_02\_LogisticBankingData.ipynb**
- Im Zusammenhang führen wir den **Cross Industry Standard Process for Data Mining (CRISP)** ein
- Details finden sich bspw. in der zitierten Quelle.
- „For business AI to succeed, you need to combine .. Machine Learning and Big Data with people who know the rules of the game in their business domain“ (Taddy, 2019, 311)

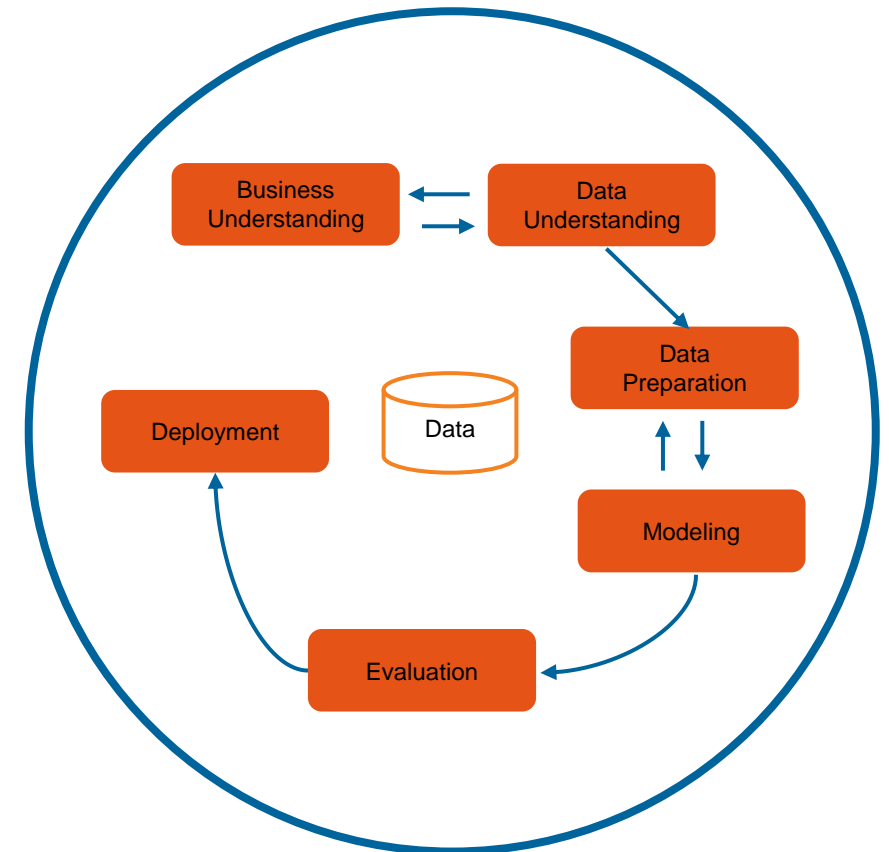
**Wo geht Domänenwissen im CRISP ein?**

**Was müssen Sie beherrschen um Teil der Lösung zu werden / sein?**



# Fallstudie BGB §489 Business Understanding

- Wir folgen dem CRISP für diese Fallstudie, bei der es um **finanzielle Selbstschädigung** geht
- Die verfremdeten Daten befinden sich in **CaseBGB489.csv**
- **Was erwarten Sie je Box des CRISP an To Dos?**
- Zum Nachlesen: „Machine Learning in der Banksteuerung – Eine Analyse der marktzinsunabhängigen Ausübung von impliziten Optionen nach BGB §489“ (mit M. Demary / S. Reuse), in „Digitale Transformation im Controlling“. Springer Gabler, Wiesbaden 2022. [https://link.springer.com/chapter/10.1007/978-3-658-38225-4\\_9](https://link.springer.com/chapter/10.1007/978-3-658-38225-4_9)



# Fallstudie BGB §489 Business Understanding

Im Fokus dieses Beispiels ist das gesetzliche Kündigungsrecht nach § 489 I Nr. 3 BGB (nachfolgend BGB-Kündigungsrecht/ -option genannt). Es findet Anwendung bei festverzinslichen Darlehen mit einer Zinsbindung von mehr als 10 Jahren, die unter keine andere Regelung zu entsprechenden Kündigungsrechten fallen (vgl. Grabbe, 2015, S. 37–79).

Durch die Kündigungssperrfrist von 10 Jahren und die Kündigungsfrist von 6 Monaten kann die Kündigung frühestens 10,5 Jahre nach Vollauszahlung des Darlehens wirksam werden. Durch § 489 IV BGB kann dieses Kündigungsrecht vertraglich nicht ausgeschlossen werden. Da es nicht auf einer expliziten Vertragsklausel beruht, handelt es sich um eine implizite Option.

Der Bank als Stillhalter steht keine Kompensation des entfallenen Gewinns zu, wie es z. B. bei der außerordentlichen Kündigung nach § 490 II BGB durch die Vorfälligkeitsentschädigung der Fall ist.

Da der Kunde das Kündigungsrecht nach Ablauf der Kündigungssperrfrist jederzeit ausüben kann, handelt es sich um eine amerikanische Option.

Quelle: Grabbe, Björn (2015): Kündigungsrechte in Privatkundendarlehen - Bewertung und Einbindung in die Geschäftsfeldsteuerung - Duisburg, Essen: DuEPublico, 2015 (Diss. Univ. Duisburg-Essen [vervielf.])

# Fallstudie BGB §489 Business Understanding

„Für den Kunden ist es nun, allgemein gesprochen, immer dann rational, sein Darlehen zu kündigen, wenn er sich entweder zu einem geringeren Zins Geld zur Ablösung des Darlehens leihen kann, oder wenn er für seine liquiden Mittel keine höher verzinsten Anlage findet.“ (Grabbe, 2015, 144).

Beides ist der Fall im vorliegenden Datensatz aus dem Jahr 2020, der aus 143 Kunden besteht.

Damit bietet sich die Wahl eines Signifikanzniveaus in Höhe von 10% an.

Man beachte, dass bei big data im Sinne „vieler Zeilen“ die p\_values ceteris paribus mit der Grösse der Stipo sinken. Deshalb passt man das Signifikanzniveau an. Ansonsten hätte man nur noch hochsignifikante Regressoren.

**Wie würden Sie es anpassen, wenn der Datensatz riesig wäre?**

Quelle: Grabbe, Björn (2015): Kündigungsrechte in Privatkundendarlehen - Bewertung und Einbindung in die Geschäftsfeldsteuerung - Duisburg, Essen: DuEPublico, 2015 (Diss. Univ. Duisburg-Essen [vervielf.])

# Fallstudie BGB §489 Data Understanding

1) Der kleinste Darlehenszinssatz beträgt 3,65% wohingegen Baufinanzierungs-konditionen im Jahr 2021 für die Restlaufzeit um 0,5% liegen.

2) Verfügt der Kunde über Mittel, so ist der Anlagezinssatz nach Steuern relevant. Eine solche Anlagealternative mit mehr als 3,65% ist schon seit Jahren nicht mehr existent.

Damit sind alle 489er Optionen im Datensatz im Geld.

**Aus didaktischen Gründen führen wir hier nur eine andere als die Publiizierte und zudem nicht optimierte Modellierung vor.**

Quelle: Grabbe, Björn (2015): Kündigungsrechte in Privatkundendarlehen - Bewertung und Einbindung in die Geschäftsfeldsteuerung - Duisburg, Essen: DuEPublico, 2015 (Diss. Univ. Duisburg-Essen [vervielf.])

# Fallstudie BGB §489 Data Understanding

Wenige Restlaufzeiten liegen im Datensatz bei knapp 10 Jahren. Damit könnte es möglicherweise noch Zeitwerte geben, die relevant sind. Das wäre aber nur der Fall, wenn die Baufinanzierungskonditionen noch Luft nach unten hätten. Davon ist nicht auszugehen, weil sich negative Refinanzierungsmöglichkeiten aus rechtlichen und betriebswirtschaftlichen Gründen nicht in Zinssätze unter Null übersetzen werden.

Aus praktischer Sicht kann das Niveau zum Untersuchungszeitpunkt 2021 zudem als Minimum betrachtet werden. **Nicht-Ausübung kann somit als irrational** aus kommerzieller Sicht bezeichnet werden, **denn der Kunde schenkt der Bank einen Geldbetrag.**

**Warum ist es wichtig, gerade solche Kund\*innen zu verstehen?**

**Wir gehen kurzzeitig zum Business Understanding zurück. Der CRISP ist nicht-linear!**

Quelle: Grabbe, Björn (2015): Kündigungsrechte in Privatkundendarlehen - Bewertung und Einbindung in die Geschäftsfeldsteuerung - Duisburg, Essen: DuEPublico, 2015 (Diss. Univ. Duisburg-Essen [vervielf.])

# Fallstudie BGB §489 Business Understanding

Es ist üblich bei binären Regressanden die Güte via der Konfusionsmatrix und mit Maßen wie der Genauigkeit usw. zu messen.

Faktisch misst dies aber nur wie treffsicher die Vorhersagen sind. Statistische Aussagen erhalten wir erst durch die vielen Tests.

Die Benchmark ist üblicherweise die Prognose des häufigsten Falls, hier also der Nicht-Ausübung. Da 77,17% nicht ausgeübt haben, wird durch diese Vorhersage eine Genauigkeit (Accuracy) von 77,17% erzielt. Wir berechnen dies anhand der Konfusionsmatrix.

Jedoch muss hier eine weitere Maßzahl neben der Genauigkeit betrachtet werden, da sich die Fehler erster und zweiter Art deutlich in ihrer Konsequenz unterscheiden.

Beim ersten Fall wird die Ausübung vorhergesagt, obwohl sie nicht erfolgt (False Positive). Die Bank hätte dann unnötigerweise einen Hedge eingekauft. Die Optionsprämie wäre verloren. Sie dürfte unterhalb 1% liegen.

Quelle: Grabbe, Björn (2015): Kündigungsrechte in Privatkundendarlehen - Bewertung und Einbindung in die Geschäftsfeldsteuerung - Duisburg, Essen: DuEPublico, 2015 (Diss. Univ. Duisburg-Essen [vervielf.])



# Fallstudie BGB §489 Business Understanding

Beim zweiten Fall (False Negative) übt ein Kunde überraschend aus.

Der entstandene Ertragsausfall lässt sich durch den Vergleich des Nominalzinses mit einer Wiederanlageopportunität approximieren.

Der durchschnittliche Nominalzins in der Stichprobe liegt bei 4,84 %. Angenommen, der aktuelle Kundenzins für ein neues Darlehen läge bei 0,50 %, dann betrüge der Schaden der Kündigung für die Bank einen Verlust von 4,34 % bzw. 4.340 € p.a. je 100.000 € Darlehenssumme.

Dies ist ein Mehrfaches der Optionsprämie. Ggf. noch Negativzinsen bei EZB.

Deshalb wird die False Negative Rate (kurz FNR) je Modell berechnet. Sie soll im Gegensatz zur Accuracy möglichst gering sein.

Bei der Benchmark beträgt sie 100%, denn es werden alle Ausüber als Nicht-Ausüber vorhergesagt. D.h. mit WS 100% werden „true positives“ als negativ prognostiziert!

# Fallstudie BGB §489 Business Understanding

Zu Beginn eines Projektes sollte auch eine Vorstellung darüber bestehen, wie die Erkenntnisse / Evidenzen in der Geschäftspraxis Eingang finden. Hier etwa:

- Auf der Ebene der Einzelgeschäfte kann bspw. eine Einbindung der Modellergebnisse im Sinne der präskriptiven Datenanalyse erfolgen, um gezielt Bestandskunden, die vermutlich ihr Kündigungsrecht rational ausüben werden, kurz vor der Ausübungsperiode mit einem Prolongationsangebot anzusprechen. Auf diese Weise wird das Risiko zwar realisiert, allerdings wird es durch die Wiederanlage im Vergleich zum vollständigen Verlust der Darlehensbeziehung mit dem Kunden begrenzt. Außerdem wird im Sinne der ganzheitlichen Kundenberatung die Kundenbeziehung durch ein Angebot, das den Kunden zu Lasten der Bank besserstellt, gestärkt.
- Durch die Verwendung eines ausreichend genauen Modells kann – unter Berücksichtigung der beschriebenen Fehlerarten – ein Preisvorteil im Wettbewerb erzielt werden, da bei marktzinsunabhängig agierenden Kunden kein Aufschlag erfolgt.

**Welche weiteren Verwendungen fallen Ihnen ein? Z B Marketingziele?**

Quelle: Grabbe, Björn (2015): Kündigungsrechte in Privatkundendarlehen - Bewertung und Einbindung in die Geschäftsfeldsteuerung - Duisburg, Essen: DuEPublico, 2015 (Diss. Univ. Duisburg-Essen [vervielf.])

# Fallstudie BGB §489 Data Understanding

Redundante Daten wurden entfernt.							
(138, 9)							
	AUSUEBUNG	SELBSTAENDIG	BERATUNGSOFFEN	ALTER	MONATSNETTO	NOMINALZINS	\
0	1.0	0.0	0.0	61.0	0.0	5.39	
1	0.0	0.0	1.0	46.0	150000.0	4.30	
2	0.0	0.0	1.0	54.0	10000.0	5.08	
3	0.0	0.0	0.0	62.0	0.0	5.00	
4	1.0	0.0	1.0	58.0	90000.0	4.95	
	GESAMTZUSAGE	URSPRUNGLAUFZEIT	RECHTVERFUEGBARSEIT				
0	90000.0	180.0	23.0				
1	70000.0	180.0	1.0				
2	120000.0	181.0	54.0				
3	70000.0	166.0	29.0				
4	460000.0	179.0	15.0				

Die Faktoren werden mit Bezeichnungen in **Grossbuchstaben** eingeführt.

Die Daten sind verfremdet, so dass keine Personen oder reale Entitäten identifiziert werden können.

Es liegt in der Natur der Sache, dass Merkmale des Entscheiders relevant sein können. Zu diesen sind im Datensatz die Datenfelder **SELBSTAENDIG** (ja = 1, auch nachfolgend) und **BERATUNGSOFFEN** (dito) von Interesse.

Wir vermuten, dass ein Selbständiger geübter ist, in finanziellen Entscheidungssituationen rational zu agieren. Zudem vermuten wir bei einem Akteur, der Beratungstermine vor Ort wahrnimmt, eine Affinität zu Finanzthemen.

**Teilen Sie diese Vermutungen?**

Quelle: Grabbe, Björn (2015): Kündigungsrechte in Privatkundendarlehen - Bewertung und Einbindung in die Geschäftsfeldsteuerung - Duisburg, Essen: DuEPublico, 2015 (Diss. Univ. Duisburg-Essen [vervielf.])

# Fallstudie BGB §489 Data Understanding

Redundante Daten wurden entfernt.							
(138, 9)							
	AUSUEBUNG	SELBSTAENDIG	BERATUNGSOFFEN	ALTER	MONATSNETTO	NOMINALZINS	\
0	1.0	0.0	0.0	61.0	0.0	5.39	
1	0.0	0.0	1.0	46.0	150000.0	4.30	
2	0.0	0.0	1.0	54.0	10000.0	5.08	
3	0.0	0.0	0.0	62.0	0.0	5.00	
4	1.0	0.0	1.0	58.0	90000.0	4.95	
	GESAMTZUSAGE	URSPRUNGLAUFZEIT	RECHTVERFUEGBARSEIT				
0	90000.0	180.0	23.0				
1	70000.0	180.0	1.0				
2	120000.0	181.0	54.0				
3	70000.0	166.0	29.0				
4	460000.0	179.0	15.0				

Daneben betrachten wir das **ALTER** in Jahren und das **MONATSNETTO**.

Weitere Merkmale werden mit Blick auf die Länge der Stichprobe nicht hin-zugefügt.

Es liegt ebenfalls in der Natur der Sache, dass Merkmale des Kredites relevant sind. Alle Kreditnehmer im Datensatz verfügen über die 489er Option.

Ihr kommerzieller Wert wird wesentlich durch die Höhe des **NOMINALZINS** [volle Prozent, z B 4,81%] und der **GESAMTZUSAGE** [in EUR] bestimmt.

**Teilen Sie diese Vermutungen?**

**Könnte es etwa bei ALTER einen nicht-linearen Zusammenhang geben ähnlich wie bei Medikamentendosierung?**

Quelle: Grabbe, Björn (2015): Kündigungsrechte in Privatkundendarlehen - Bewertung und Einbindung in die Geschäftsfeldsteuerung - Duisburg, Essen: DuEPublico, 2015 (Diss. Univ. Duisburg-Essen [vervielf.])

# Fallstudie BGB §489 Data Understanding

Redundante Daten wurden entfernt.

(138, 9)

	AUSUEBUNG	SELBSTAENDIG	BERATUNGSOFFEN	ALTER	MONATSNETTO	NOMINALZINS	\
0	1.0	0.0	0.0	61.0	0.0	5.39	
1	0.0	0.0	1.0	46.0	150000.0	4.30	
2	0.0	0.0	1.0	54.0	10000.0	5.08	
3	0.0	0.0	0.0	62.0	0.0	5.00	
4	1.0	0.0	1.0	58.0	90000.0	4.95	

	GESAMTZUSAGE	URSPRUNGSLAUFZEIT	RECHTVERFUEGBARSEIT
0	90000.0	180.0	23.0
1	70000.0	180.0	1.0
2	120000.0	181.0	54.0
3	70000.0	166.0	29.0
4	460000.0	179.0	15.0

Je höher der Nominalzins umso grösser der innere Wert der Option.

Je höher die Gesamtzusage desto grösser der absolute Wert der Optionsposition.

**Welche Vorzeichen vermuten wir bzgl. der Koeffizienten der Faktoren?**

**In welche Richtung wirken Sie? Je höher desto (un-)wahrscheinlicher die AÜ?**

Daneben betrachten wir weitere Attribute, die möglicherweise Einfluss haben können, bei denen wir aber keine Vermutungen über die Wirkungsrichtung anstellen möchten.

Diese Attribute sind **URSPRUNGSLAUFZEIT** [in Monaten] und **RECHTVERFUEGBARSEIT**, d.h. der Zeitraum in Monaten seitdem eine Ausübung möglich ist.

Quelle: Grabbe, Björn (2015): Kündigungsrechte in Privatkundendarlehen - Bewertung und Einbindung in die Geschäftsfeldsteuerung - Duisburg, Essen: DuEPublico, 2015 (Diss. Univ. Duisburg-Essen [vervielf.])

# Fallstudie BGB §489 Data Understanding

Redundante Daten wurden entfernt.

(138, 9)

	AUSUEBUNG	SELBSTAENDIG	BERATUNGSOFFEN	ALTER	MONATSNETTO	NOMINALZINS	\
0	1.0	0.0	0.0	61.0	0.0	5.39	
1	0.0	0.0	1.0	46.0	150000.0	4.30	
2	0.0	0.0	1.0	54.0	10000.0	5.08	
3	0.0	0.0	0.0	62.0	0.0	5.00	
4	1.0	0.0	1.0	58.0	90000.0	4.95	

	GESAMTZUSAGE	URSPRUNGSLAUFZEIT	RECHTVERFUEGBARSEIT
0	90000.0	180.0	23.0
1	70000.0	180.0	1.0
2	120000.0	181.0	54.0
3	70000.0	166.0	29.0
4	460000.0	179.0	15.0

Variablen	Beschreibung
<b>Ausübung (Y):</b>	y = Ausübung 0,1 (0 = keine Ausübung / 1 = Ausübung) = binär codiert
Selbstständig	Ausprägung: 0,1 (0 = Angestellter / 1 = Selbstständig) = binär codiert
Beratungsoffen	Ausprägung: 0,1 (0 = lehnt Beratung ab / 1 = offen für Beratung) = binär codiert
Alter	Alter des Kreditnehmers
Monatsnetto	Monatliches Gehalt des Kreditnehmers (Netto)
Nominalzins	Zinssatz des Darlehens
Gesamtzusage	Darlehenshöhe
Ursprungslaufzeit	Laufzeit des Darlehens
Rechtsverfügbarkeit	Laufzeit, seitdem eine Ausübung des Rechts möglich ist

# Fallstudie BGB §489 Data Understanding

Mit (ausnahmsweise) R lassen sich folgende Visualisierungen erzeugen

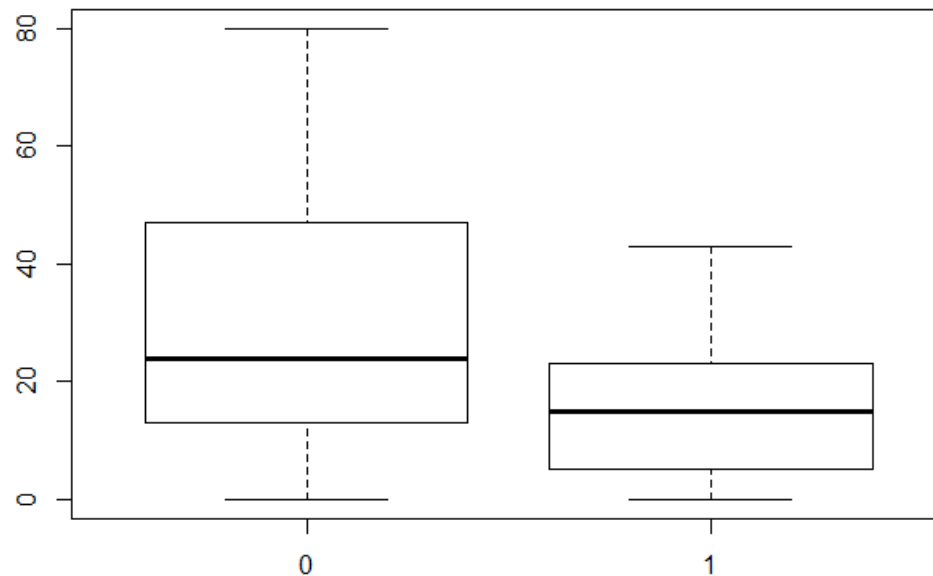
Redundante Daten wurden entfernt.

(138, 9)

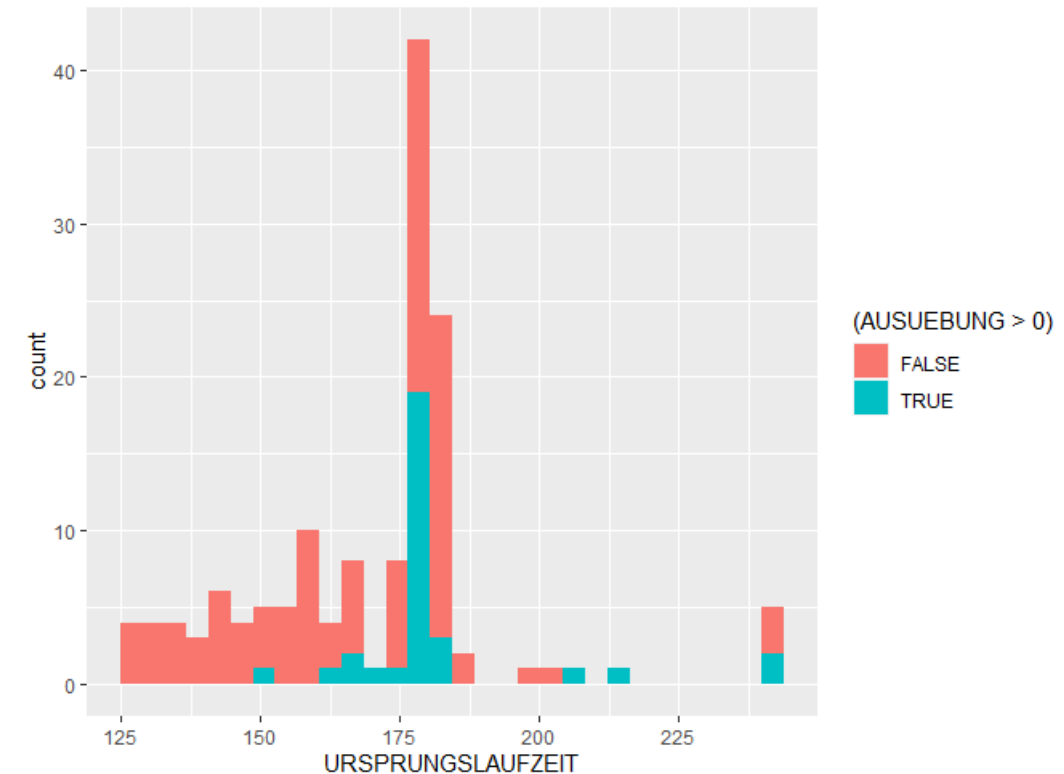
	AUSUEBUNG	SELBSTAENDIG	BERATUNGSOFFEN	ALTER	MONATSNETTO	NOMINALZINS	\
0	1.0	0.0	0.0	61.0	0.0	5.39	
1	0.0	0.0	1.0	46.0	150000.0	4.30	
2	0.0	0.0	1.0	54.0	10000.0	5.08	
3	0.0	0.0	0.0	62.0	0.0	5.00	
4	1.0	0.0	1.0	58.0	90000.0	4.95	

	GESAMTZUSAGE	URSPRUNGLAUFZEIT	RECHTVERFUEGBARSEIT
0	90000.0	180.0	23.0
1	70000.0	180.0	1.0
2	120000.0	181.0	54.0
3	70000.0	166.0	29.0
4	460000.0	179.0	15.0

RECHTVERFUEGBARSEIT vs. AUSUEBUNG



Was erkennen Sie?



# Fallstudie BGB §489 Business Understanding

**Man erkennt, dass die Mehrheit nicht ausgeübt hat**

**Gerundet 80% haben damit ihrer Bank sehr viel Geld geschenkt**

Derartige, suboptimale Ausübung von Optionen ist keine Seltenheit:

„In our last validation exercise, we show that .. investors are less likely to exercise their [equity] options optimally”

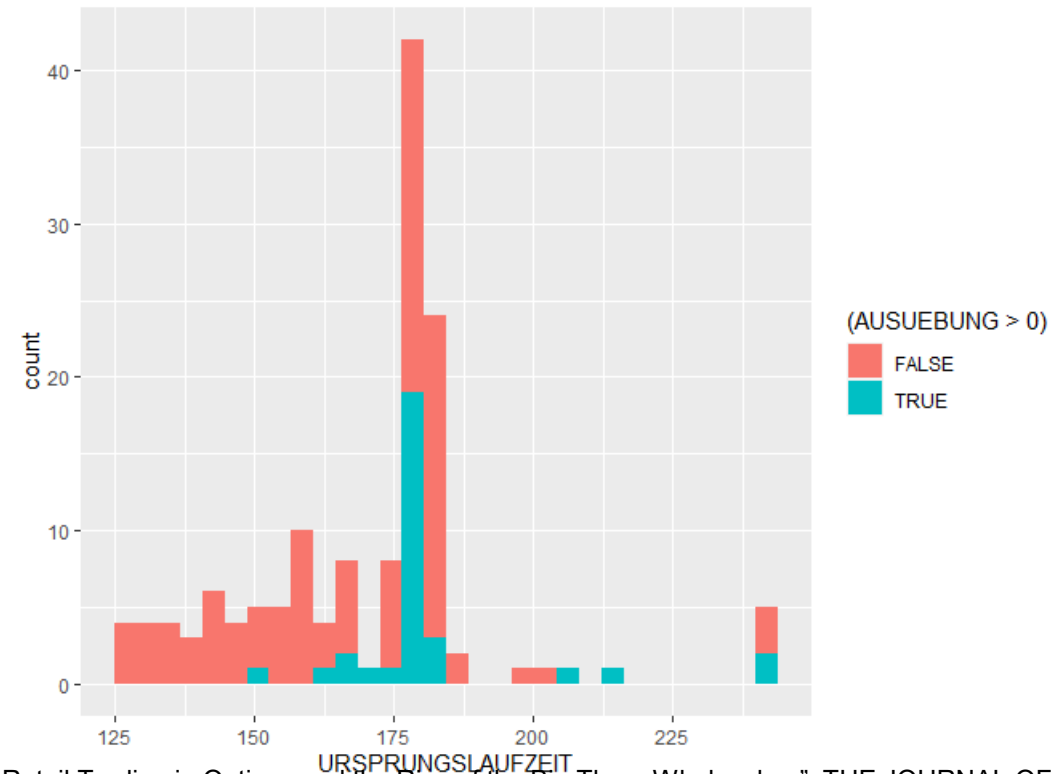
“Previous research documents that not all American options are exercised rationally (e.g., Poteshman and Serbin (2003)). Focusing on early exercise decisions, Battalio, Figlewski, and Neal (2020), Cosma et al. (2020), Jensen and Pedersen (2016), and Barraclough and Whaley (2012) show in more recent data that a fraction of investors still fail to exercise their options optimally.”

Redundante Daten wurden entfernt.

(138, 9)

	AUSUEBUNG	SELBSTAENDIG	BERATUNGSOFFEN	ALTER	MONATSNETTO	NOMINALZINS	\
0	1.0	0.0	0.0	61.0	0.0	5.39	
1	0.0	0.0	1.0	46.0	150000.0	4.30	
2	0.0	0.0	1.0	54.0	10000.0	5.08	
3	0.0	0.0	0.0	62.0	0.0	5.00	
4	1.0	0.0	1.0	58.0	90000.0	4.95	

	GESAMTZUSAGE	URSPRUNGLAUFZEIT	RECHTVERFUEGBARSEIT
0	90000.0	180.0	23.0
1	70000.0	180.0	1.0
2	120000.0	181.0	54.0
3	70000.0	166.0	29.0
4	460000.0	179.0	15.0



Quelle: SVETLANA BRYZGALOVA, ANNA PAVLOVA, and TAISIYA SIKORSKAYA, 2023, „Retail Trading in Options and the Rise of the Big Three Wholesalers”, THE JOURNAL OF FINANCE • VOL. 78, NO. 6 • DECEMBER 2023



# Fallstudie BGB §489 Modeling

Wir beginnen mit Logit

**Interpretieren Sie!**

**Achten Sie auch auf die Vorzeichen!**

Optimization terminated successfully.

Current function value: 0.313769

Iterations 8

## Logit Regression Results

Dep. Variable:	AUSUEBUNG	No. Observations:	124			
Model:	Logit	Df Residuals:	115			
Method:	MLE	Df Model:	8			
Date:	Mon, 07 Aug 2023	Pseudo R-squ.:	0.4012			
Time:	12:45:49	Log-Likelihood:	-38.907			
converged:	True	LL-Null:	-64.980			
Covariance Type:	nonrobust	LLR p-value:	1.577e-08			
=====						
	coef	std err	z	P> z	[0.025	0.975]
-----						
const	-19.8710	5.519	-3.601	0.000	-30.687	-9.055
SELBSTAENDIG	1.5052	0.723	2.081	0.037	0.087	2.923
BERATUNGSOFFEN	1.4906	0.686	2.173	0.030	0.146	2.835
ALTER	-0.0350	0.028	-1.242	0.214	-0.090	0.020
MONATSNETTO	-1.212e-05	5.6e-06	-2.163	0.031	-2.31e-05	-1.14e-06
NOMINALZINS	1.5677	0.705	2.224	0.026	0.186	2.949
GESAMTZUSAGE	-9.179e-07	8.98e-07	-1.022	0.307	-2.68e-06	8.42e-07
URSPRUNGSLAUFZEIT	0.0794	0.019	4.173	0.000	0.042	0.117
RECHTVERFUEGBARSEIT	-0.0639	0.021	-3.115	0.002	-0.104	-0.024
=====						

Optimization terminated successfully.  
Current function value: 0.313769  
Iterations 8

```
=====
Logit Regression Results
=====
Dep. Variable:      AUSUEBUNG      No. Observations:      124
Model:              Logit          Df Residuals:           115
Method:             MLE            Df Model:              8
Date:               Mon, 07 Aug 2023 Pseudo R-squ.:             0.4012
Time:               12:45:49        Log-Likelihood:         -38.907
converged:          True           LL-Null:               -64.980
Covariance Type:    nonrobust      LLR p-value:           1.577e-08
=====
                    coef    std err          z      P>|z|      [0.025    0.975]
-----
const              -19.8710     5.519     -3.601    0.000    -30.687    -9.055
SELBSTAENDIG        1.5052     0.723     2.081    0.037     0.087     2.923
BERATUNGSOFFEN      1.4906     0.686     2.173    0.030     0.146     2.835
ALTER              -0.0350     0.028    -1.242    0.214    -0.090     0.020
MONATSNETTO         -1.212e-05    5.6e-06    -2.163    0.031    -2.31e-05    -1.14e-06
NOMINALZINS         1.5677     0.705     2.224    0.026     0.186     2.949
GESAMTZUSAGE        -9.179e-07    8.98e-07    -1.022    0.307    -2.68e-06    8.42e-07
URSPRUNGSLAUFZEIT   0.0794     0.019     4.173    0.000     0.042     0.117
RECHTVERFUEGBARSEIT -0.0639     0.021    -3.115    0.002    -0.104    -0.024
=====
```

# Fallstudie BGB §489 Modeling

Wir beginnen mit Logit

Interpretieren Sie!

Welche Regressoren würden Sie entfernen / überarbeiten?

Achten Sie auch auf die Vorzeichen!

```
##      X      AUSUEBUNG      SELBSTAENDIG      BERATUNGSOFFEN
## Min.   : 1.0      Min.   :0.0000      Min.   :0.0000      Min.   :0.0000
## 1st Qu.: 36.5      1st Qu.:0.0000      1st Qu.:0.0000      1st Qu.:0.0000
## Median : 72.0      Median :0.0000      Median :0.0000      Median :0.0000
## Mean   : 72.0      Mean   :0.2238      Mean   :0.1399      Mean   :0.4196
## 3rd Qu.:107.5      3rd Qu.:0.0000      3rd Qu.:0.0000      3rd Qu.:1.0000
## Max.   :143.0      Max.   :1.0000      Max.   :1.0000      Max.   :1.0000
##      ALTER      MONATSNETTO      NOMINALZINS      GESAMTZUSAGE
## Min.   :16.00      Min.   : 0      Min.   :3.650      Min.   : 20000
## 1st Qu.:52.00      1st Qu.: 0      1st Qu.:4.500      1st Qu.: 80000
## Median :56.00      Median :10000      Median :4.870      Median :120000
## Mean   :57.24      Mean   :33427      Mean   :4.807      Mean   :233916
## 3rd Qu.:63.00      3rd Qu.:30000      3rd Qu.:5.080      3rd Qu.:210000
## Max.   :84.00      Max.   :370000      Max.   :5.853      Max.   :2000000
## URSPRUNGSLAUFZEIT RECHTVERFUEGBARSEIT
## Min.   :127.0      Min.   : 0.00
## 1st Qu.:157.0      1st Qu.:10.50
## Median :178.0      Median :22.00
## Mean   :170.4      Mean   :25.02
## 3rd Qu.:180.0      3rd Qu.:40.00
## Max.   :242.0      Max.   :80.00

dfrm_KUNDE = dfrm_clean[, -1]
```

# Fallstudie BGB §489 Modeling

Die Regressoren A) URSPRUNGSLAUFZEIT und B) RECHTVERFUEGBARSEIT erscheinen ad hoc hoch korreliert. Der Korrelationskoeffizient liegt jedoch unter 50%.

Die erste Variable gibt an, wie lange man noch haben könnte, die Zweite wie lange man schon „geschlafen“ hat. Die Darlehen laufen verschieden lange. Manches hatte ursprünglich 15, ein anderes 20 Jahre. Deshalb leiten sich die Größen nicht 1:1 voneinander ab.

Zeitstrahl: 10. Jahr --B-- heute --A-- Laufzeitende

# Fallstudie BGB §489 Modeling

Für das MLP müssen wir diesmal zwei Datenvorverarbeitungsfunktionen nutzen, weil der Regressand  $y$  vom Typ long sein muss

## Was ist unklar?

```
# Standardisierung
scaler = StandardScaler()
scaler.fit(X_train)
X_train_scaled = scaler.fit_transform(X_train)
X_test_scaled = scaler.transform(X_test)
```

```
#prep torch data
t_X_train = func_turn_df_into_t_float(X_train_scaled)
t_X_valid = func_turn_df_into_t_float(X_test_scaled)

t_y_train = func_turn_df_into_t_long(y_train)
t_y_valid = func_turn_df_into_t_long(y_test)
```

```
myHIDDENNone = 6 #we have 8 dim input + 1 out = 9, two thirds = 6
myMAX_EPOCHS = 500 #500 #more than 2000 does not pay off, valid
myLEARNING_RATE = 0.1
myBATCH_SIZE = 32 #32 is min
```

```
def func_turn_df_into_t_float(df_arg):
    t_ret = torch.from_numpy(np.array(df_arg))
    t_ret = t_ret.to(torch.float)
    return t_ret

def func_turn_df_into_t_long(df_arg):
    t_ret = torch.from_numpy(np.array(df_arg))
    t_ret = t_ret.to(torch.long)
    return t_ret
```

# Fallstudie BGB §489 Modeling

Diesmal arbeiten wir mit einer anderen Fehlerfunktion, der **Cross Entropy** (CE):

```
loss_fn = nn.CrossEntropyLoss()
```

Die CE misst den Abstand zwischen zwei Verteilungen. Sie ist minimal, wenn nur diese Paarungen auftreten, also das Modell korrekt vorhersagt:

Realität  $y=1$  und Vorhersage  $p$  nahe 1  $\Rightarrow \log(p) = 0$ , egal welche Basis, also ob  $\ln$  oder  $\log_2$ . Produkt ist 0.

Realität  $y=0$  und Vorhersage  $p$  nahe 0  $\Rightarrow \log(p) \ll 0$ , egal welche Basis, also ob  $\ln$  oder  $\log_2$ . Produkt ist 0.

## Loss Function: Binary Cross-Entropy / Log Loss

If you look this loss function up, this is what you'll find:

$$H_p(q) = -\frac{1}{N} \sum_{i=1}^N y_i \cdot \log(p(y_i)) + (1 - y_i) \cdot \log(1 - p(y_i))$$

Binary Cross-Entropy / Log Loss

where  $y$  is the label (1 for green points and 0 for red points) and  $p(y)$  is the predicted probability of the point being green for all  $N$  points.

Quelle: <https://towardsdatascience.com/understanding-binary-cross-entropy-log-loss-a-visual-explanation-a3ac6025181a>, Zugriff 8 Jan 2022

# Fallstudie BGB §489 Modeling

## Konkretes Zahlenbsp.

Y=Real	p=Prognose	log(p)	Y-forecast	Summand für dieses i
1	0,00001	-5,0000	0	5,0000
1	0,4	-0,3979	0	0,3979
<b>1</b>	<b>0,6</b>	<b>-0,2218</b>	<b>1</b>	<b>0,2218</b>
<b>1</b>	<b>0,999999</b>	<b>0,0000</b>	<b>1</b>	<b>0,0000</b>
<b>0</b>	<b>0,00001</b>	<b>-5,0000</b>	<b>0</b>	<b>0,0000</b>
<b>0</b>	<b>0,4</b>	<b>-0,3979</b>	<b>0</b>	<b>0,2218</b>
0	0,6	-0,2218	1	0,3979
0	0,999999	0,0000	1	6,0000

**Korrekte Vorhersagen** führen zu kleinen Summanden

Minimierung der Fehlerfunktion also abermals zielführend (statt KQ, kleinste CE)

## Loss Function: Binary Cross-Entropy / Log Loss

If you look this loss function up, this is what you'll find:

$$H_p(q) = -\frac{1}{N} \sum_{i=1}^N y_i \cdot \log(p(y_i)) + (1 - y_i) \cdot \log(1 - p(y_i))$$

Binary Cross-Entropy / Log Loss

where **y** is the label (1 for green points and 0 for red points) and **p(y)** is the predicted probability of the point being green for all N points.

Die CE lässt sich unter die Kullback-Leibler Divergenz subsumieren, mit der die Distanz zwischen zwei Verteilungen gemessen wird. Hierzu Goodfellow, 2016, 72.

Quelle: <https://towardsdatascience.com/understanding-binary-cross-entropy-log-loss-a-visual-explanation-a3ac6025181a>, Zugriff 8 Jan 2022

# Fallstudie BGB §489 Modeling

Bis auf die Fehlerfunktion ist es wie üblich

**Was ist unklar?**

```
def generic_training_loop_extended(n_epochs, optimizer, model, loss_fn, train_loader, val_loader):
    for epoch in range(1, n_epochs + 1):
        loss_train_avg = []
        for step, (batch_x, batch_y) in enumerate(train_loader):
            b_x = Variable(batch_x)
            b_y = Variable(batch_y)
            train_t_p = model(b_x) #all details are in model
            train_loss = loss_fn(train_t_p, b_y) #and the loss function
            loss_train_avg.append(train_loss.item())
            optimizer.zero_grad()
            train_loss.backward()
            optimizer.step()
        with torch.no_grad():
            val_t_p = model(val_loader.data_loader.get_batch(0))
            val_loss = loss_fn(val_t_p, val_loader.data_loader.get_batch(1))
            losses_val.append(val_loss.item())
            assert val_loss.requires_grad == False
        losses_train.append(train_loss.item())
        if epoch <= 100 or epoch % 50 == 0:
            print("Epoch", {epoch}, f"Avg. training loss over all batches: {np.average(loss_train_avg[0:step]):.4f}",
                  f"Validation loss: {val_loss.item():.4f}",
                  "\n \tLast batch out of", {step}, f"has training loss: {train_loss.item():.4f}")

    return
```

*#PyTorch MLP*

```
torch_dataset = Data.TensorDataset(t_X_train, t_y_train)
train_loader = Data.DataLoader(
    dataset=torch_dataset,
    batch_size=myBATCH_SIZE,
    shuffle=True, num_workers=1,) #only one subprocess needed
```

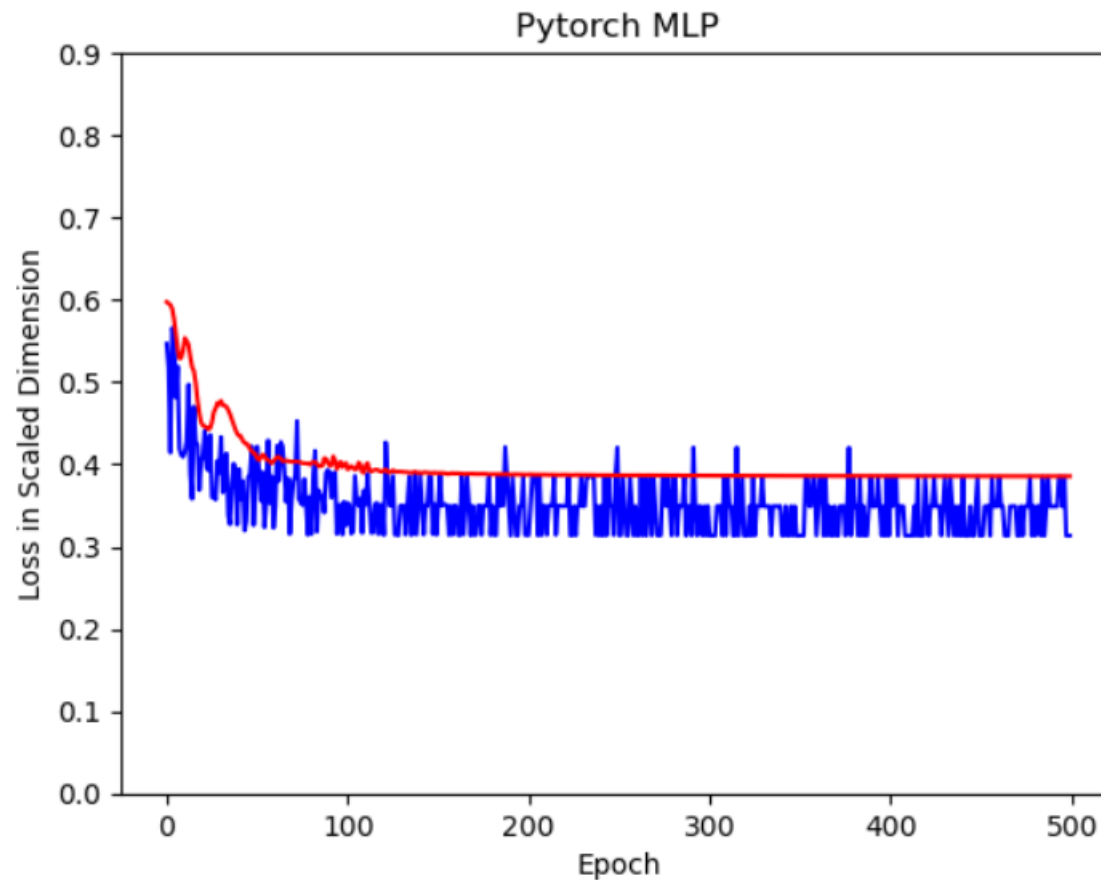
```
model_pt = nn.Sequential(nn.Linear(8, myHIDDENNone),
                          nn.Sigmoid(),
                          nn.Linear(myHIDDENNone, 2), nn.Softmax(dim=1))
```

```
loss_fn = nn.CrossEntropyLoss() #watch this!
optimizer = torch.optim.Adam(model_pt.parameters(), lr=myLEARNING_RATE)
```

# Fallstudie BGB §489 Modeling

Nach 500 Epochen tut sich nichts mehr

Der Validierungsfehler bildet gegen Ende eine obere Schranke

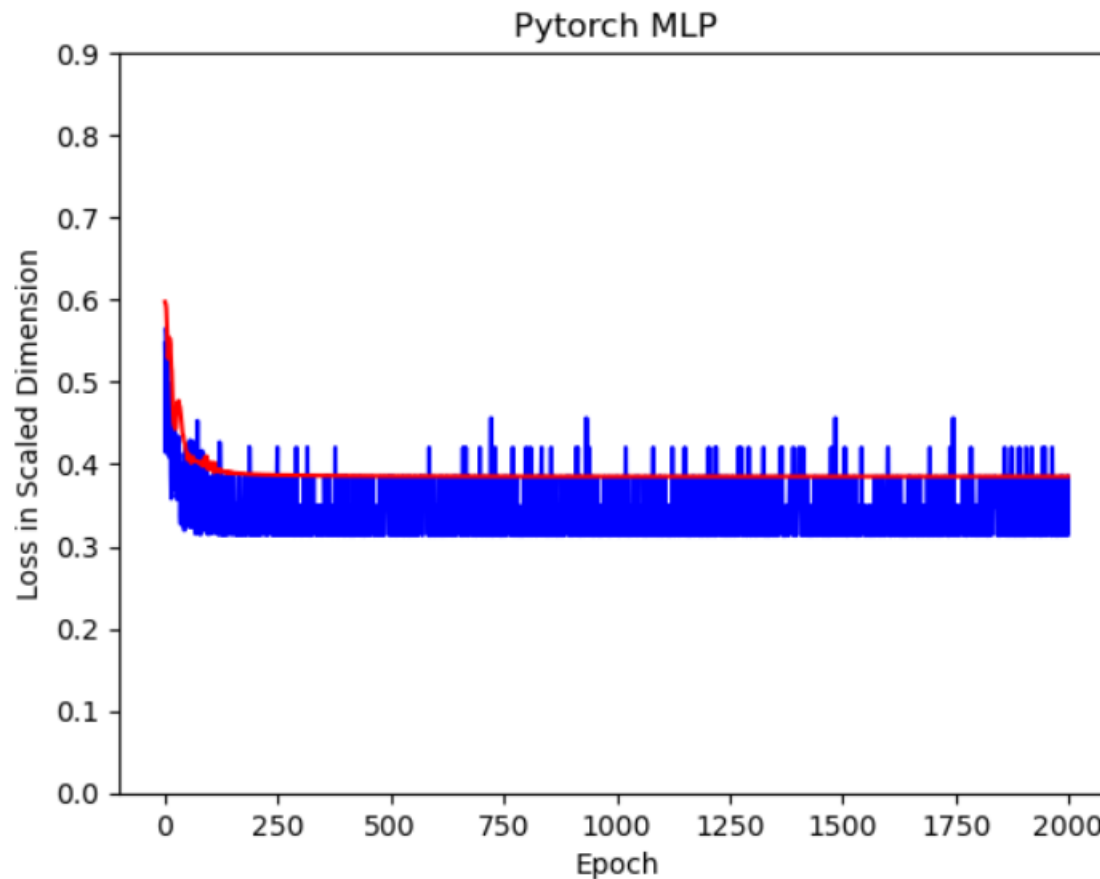




# Fallstudie BGB §489 Modeling

Auch ein sehr langes Netztraining ändert nicht viel

Der Validierungsfehler wird ab Schritt 500 zuweilen wieder überschritten



COMPARISON ON TEST SET=====

Number of exercises in test data 4.0

As percentage 0.286

Benchmark Accuracy on test data 0.714

Accuracy Logit on test data: 0.643

Confusion Matrix Logit on test data:

```
[[9 1]
```

```
[4 0]]
```

Accuracy MLP on test data: 0.929

Confusion Matrix MLP on test data:

```
[[9 1]
```

```
[0 4]]
```

# Fallstudie BGB §489 Modeling

Interessant sind die Vorzeichen

Das MLP arbeitet auf skalierten Daten!

```
#inspect model
ig = IntegratedGradients(model_pt)#Specifically, integrated g
#are defined as the path integral of the gradients along the
#see https://arxiv.org/pdf/1703.01365.pdf
ig_input_tensor = t_X_train
ig_input_tensor.requires_grad_()
attr, delta = ig.attribute(ig_input_tensor, target=0, return_c
#target class No 0 because we only have one output dimension
attr = attr.detach().numpy()
np.round(attr,2)
importances = np.mean(attr, axis=0)
feature_names = list(X_train.columns)
for i in range(len(feature_names)):
    print(feature_names[i], ": ", '%.3f'%(importances[i]))
```

```
SELBSTAENDIG : -0.042
BERATUNGSOFFEN : 0.018
ALTER : -0.011
MONATSNETTO : -0.025
NOMINALZINS : -0.083
GESAMTZUSAGE : 0.029
URSPRUNGSLAUFZEIT : -0.041
RECHTVERFUEGBARSEIT : -0.047
```

Optimization terminated successfully.

Current function value: 0.313769

Iterations 8

## Logit Regression Results

Dep. Variable:	AUSUEBUNG	No. Observations:	124
Model:	Logit	Df Residuals:	115
Method:	MLE	Df Model:	8
Date:	Mon, 07 Aug 2023	Pseudo R-squ.:	0.4012
Time:	12:45:49	Log-Likelihood:	-38.907
converged:	True	LL-Null:	-64.980
Covariance Type:	nonrobust	LLR p-value:	1.577e-08

	coef	std err	z	P> z	[0.025	0.975]
const	-19.8710	5.519	-3.601	0.000	-30.687	-9.055
SELBSTAENDIG	1.5052	0.723	2.081	0.037	0.087	2.923
BERATUNGSOFFEN	1.4906	0.686	2.173	0.030	0.146	2.835
ALTER	-0.0350	0.028	-1.242	0.214	-0.090	0.020
MONATSNETTO	-1.212e-05	5.6e-06	-2.163	0.031	-2.31e-05	-1.14e-06
NOMINALZINS	1.5677	0.705	2.224	0.026	0.186	2.949
GESAMTZUSAGE	-9.179e-07	8.98e-07	-1.022	0.307	-2.68e-06	8.42e-07
URSPRUNGSLAUFZEIT	0.0794	0.019	4.173	0.000	0.042	0.117
RECHTVERFUEGBARSEIT	-0.0639	0.021	-3.115	0.002	-0.104	-0.024

# Fallstudie BGB §489 Evaluation

Als Erstes müssen die **Benchmarks** bestimmt werden

Naive Modelle, die stets den häufigsten Fall vorhersagen, erreichen schon gute Genauigkeiten

Test- und Trainingsdaten sind ähnlich genug, also sinnvoll

Das **Logit Modell** kann zwar direkt interpretiert werden, jedoch sagt es weniger gut vorher. Bspw. Matrix oben rechts 7 Fälle, die als false = 0 prognostiziert wurden, obwohl sie eine 1 (positiv) waren

Besonders diese **7+1** (zweite Matrix) **false negatives** würden praktisch wehtun

Das **MLP** hat deutlich mehr Modellparameter als das Logit

Deswegen kann es sich besser anpassen

Es gibt nur **1+1 false negatives**

Die Genauigkeit ist gleichbleibend hoch

```
Number of exercises in test data 4.0
As percentage 0.286
Benchmark Accuracy on train data 0.782
Benchmark Accuracy on test data 0.714
```

```
Realität: 0    1
Modellprognose
0
1
Accuracy Logit on train data: 0.847
Confusion Matrix Logit on train data:
[[90  7]
 [12 15]]
Accuracy Logit on test data: 0.643
Confusion Matrix Logit on test data:
[[9 1]
 [4 0]]

Accuracy MLP on train data: 0.968
Confusion Matrix MLP on train data:
[[96  1]
 [ 3 24]]
Accuracy MLP on test data: 0.929
Confusion Matrix MLP on test data:
[[9 1]
 [0 4]]
```

# Fallstudie BGB §489 Deloyment

Wir haben den CRISP einmal durchlaufen und würden nun wieder zurückgehen

Für eine praktische Anwendung würde man unsinnige Variablen entfernen

Bei einem MONATSNETTO von Null ist nicht von Kreditwürdigkeit auszugehen ...

Ähnlich wie bei Logit können auch beim MLP Hypothesentests für einzelne Gewichte und damit Regressoren durchgeführt werden

JEDOCH benötigen diese Theoreme GENAU wie MaxLik GROSSE STICHPROBEN

Dies ist hier definitiv nicht der Fall

DESHALB würde man Daten von mehreren Banken poolen

Dabei muss die Datenqualität jedoch einheitlich sein, weil sonst eine Korrelation der Störterme für Daten aus Banken mit schlechter IT resultiert

Damit wäre eine Annahme für die Konsistenz verletzt

ANGENOMMEN all das ist geleistet, dann gilt es das MLP zu erklären

# Fallstudie BGB §489 Deployment: Gesellschaftliche Debatte

Vor dem Praxiseinsatz lohnt ein Innehalten

**Lesen Sie diesen Abstract aus dem JoF**

**Warum lohnt es? Was könnten die Konsequenzen Ihres Handelns sein?**

**Freiwillige vor: Nutzen Sie StrokeData.csv und PYTPRA\_03\_03\_LogisticMedicalData.ipynb um ein besseres Modell als die naive Vorhersage zu bauen!**

*The Journal of* **FINANCE**  
*The Journal of* THE AMERICAN FINANCE ASSOCIATION

THE JOURNAL OF FINANCE • VOL. LXXVII, NO. 1 • FEBRUARY 2022

## **Predictably Unequal? The Effects of Machine Learning on Credit Markets**

ANDREAS FUSTER, PAUL GOLDSMITH-PINKHAM, TARUN RAMADORAI,  
and ANSGAR WALTHER

### **ABSTRACT**

Innovations in statistical technology in functions including credit-screening have raised concerns about distributional impacts across categories such as race. Theoretically, distributional effects of better statistical technology can come from greater flexibility to uncover structural relationships or from triangulation of otherwise excluded characteristics. Using data on U.S. mortgages, we predict default using traditional and machine learning models. We find that Black and Hispanic borrowers are disproportionately less likely to gain from the introduction of machine learning.

# Fallstudie BGB §489 Deployment: EU AI Act

**Annahme:** Wir nutzen das Modell, um Bestandskunden, die vermutlich ihr Kündigungsrecht rational ausüben werden, kurz vor der Ausübungsperiode mit einem Prolongationsangebot anzusprechen.

**Stellt das Modell zur Vorhersage der Optionsausübung KI dar?**

## Vertrauenswürdige KI

Umsetzung des EU AI Act als Value Treiber

”

[Ein] „KI-System“ [ist] ein maschinengestütztes System, das für einen in wechselndem Maße autonomen Betrieb ausgelegt sind, das nach seiner Einführung anpassungsfähig sein kann und das aus den erhaltenen Eingaben für explizite oder implizite Ziele ableitet, wie Ergebnisse wie etwa Vorhersagen, Inhalte, Empfehlungen oder Entscheidungen hervorgebracht werden, die physische oder virtuelle Umgebungen beeinflussen können.<sup>6</sup>

“

Quelle: PWC WP Gesellschaft, White Paper Vertrauenswürdige KI, März 2024

# Fallstudie BGB §489 Deployment: EU AI Act

**In welcher Kategorie  
würden Sie das Modell  
einordnen?**

## Hochrisiko-KI-Systeme nach Annex I (vormals Annex II)

**Abschnitt A** – Hochrisiko-KI-Systeme für die, zur Vermeidung von doppelter regulatorischer Belastung, Flexibilität bei der Umsetzung der Compliance eingeräumt wird:

- |   |   |  |
|---|---|--|
| • Maschinen                                   | • Geräte und Schutzsysteme in explosionsgefährdeten Bereichen | • Persönliche Schutzausrüstungen                 |
| • Sicherheit von Spielzeugen                  | • Funkanlagen   | • Geräte zur Verbrennung gasförmiger Brennstoffe |
| • Sportboote und Wassermotorräder             | • Druckgeräte   | • Medizinprodukte                                |
| • Aufzüge und Sicherheitsbauteile für Aufzüge | • Seilbahnen  | • In-vitro-Diagnostika                           |

**Abschnitt B** – Hochrisiko-KI-Systeme, die sektoral reguliert, jedoch von den meisten Anforderungen ausgenommen sind:

- |  |  |   |
|--|--|---|
| • (Sicherheit in der) Zivilluftfahrt       | • Schiffsausrüstung  | selbstständige technische Einheiten für diese Fahrzeuge |
| • Zwei-, drei- und vierrädrige Fahrzeuge   | • Interoperabilität des Eisenbahnsystems                               |   |
| • Land- und forstwirtschaftliche Fahrzeuge | • Kraftfahrzeuge und Kraftfahrzeuganhänger sowie Systeme, Bauteile und |   |

## Hochrisiko-KI-Systeme nach Annex III

- |                           |  |   |
|---------------------------|--|---|
| • Biometrische Systeme    | • Essenzielle private und öffentliche Dienstleistungen inklusive Finanzen und Versicherungen | • Migration, Asyl und Grenzkontrollen     |
| • Kritische Infrastruktur | • Strafverfolgung  | • Rechtspflege und demokratische Prozesse |
| • Bildungswesen           |  |   |
| • Personalwesen           |  |   |

Quelle: PWC WP Gesellschaft, White Paper Vertrauenswürdige KI, März 2024



# Fallstudie BGB §489 Deployment: EU AI Act

Diese Anforderungen sind nicht die Einzigen:

Beispielhaft für eine Überschneidung bestehender Compliance-Prozesse mit den künftigen Regularien des EU AI Act ist die Bewertung von Risiken: Während der EU AI Act eine Bewertung von zu erwartenden Risiken für die Gesundheit, Sicherheit oder die fundamentalen Rechte vorschreibt, ist nach der DSGVO eine „Datenschutz-Folgenabschätzung“ bei voraussichtlich hohen Risiken für die Rechte und Freiheiten Betroffener vorzunehmen. Aus Corporate-Social-Governance-Gesichtspunkten ist zudem

**Welche Anreize setzt all dies zur Erbringung essenzieller Finanzdienstleistungen?**

**Effekte bzgl. Produktionskosten & Bürokratieabbau?**

## Anforderungen an Hochrisiko-KI-Systeme

Die Anforderungen des EU AI Act für Hochrisiko-KI-Systeme sind umfangreich und stellen Organisationen vor komplexe Herausforderungen. In der folgenden Übersicht sind die

zentralen Anforderungsbereiche für Hochrisiko-KI-Systeme nach dem EU AI Act dargestellt:

### Ressourcenmanagement



Bereitstellung,  
Zuteilung,  
Kapazität,  
Verantwortung

### Risikomanagement



Identifikation,  
Bewertung,  
Reduktion,  
Überwachung

### Datenmanagement



Qualitätsbewertung,  
Annotation, Korrektur,  
Logging, Datenteilung,  
Datenschutz

### Lebenszyklus-Management



Entwicklung & Betrieb,  
Leistung, Tests,  
Robustheit, Überwachung,  
Cybersicherheit

### Transparenz



Bedarf,  
Informationen,  
Instruktionen,  
Erklärbarkeit

### Dokumentation



Model Cards,  
Data Sheets,  
System Information,  
Aufbewahrung

### Konformitätsbewertung



Prüfung,  
Erklärung,  
CE-Kennzeichnung,  
Korrekturen

### Interaktion mit Behörden



Registrierung,  
Kommunikation,  
Zugang,  
Berichtswesen

Quelle: PWC WP Gesellschaft, White Paper Vertrauenswürdige KI, März 2024



# Fallstudie BGB §489 Deployment: EU AI Act

Wir illustrieren einzelne Anforderungen:

**Was davon würde man sowieso tun (müssen)?**

**Was ist Zusatzaufwand?**

**Was davon könnte eine Art „Dealbreaker“ (Deployment Breaker) darstellen?**

## Datenmanagement

Hochwertige Daten sind zur Entwicklung von guten KI-Systemen unerlässlich. Beim Datenmanagement geht es im EU AI Act daher primär darum, geeignete Schritte vorzunehmen, um eine hohe Datenqualität zu erreichen und Daten vor dem Zugriff durch unberechtigte Personen zu schützen.

Eine enorme Herausforderung stellt die Verarbeitung von Datensätzen während der Entwicklung und Nutzung dar. So sieht der EU AI Act spezifische und umfassende Qualitätskriterien und -sicherungsmaßnahmen vor. Daraus folgt eine Reihe von Anforderungen: Datensätze müssen regelmäßig und transparent aufbereitet werden. Organisationen müssen zulässige und gegebenenfalls gewünschte Abweichungen ermitteln und definieren. Sie müssen die Verfügbarkeit, Menge und Eignung der Datensätze sowie mögliche (Ergebnis-)Verzerrungen prüfen. Und zu guter Letzt mögliche Datenlücken oder Mängel ermitteln. Der EU AI Act konkretisiert diese Anforderungen jedoch nicht weiter, sodass Betroffene das rechtliche Mindestmaß für das jeweils in Rede stehende Hochrisiko-KI-System selbst ermitteln, dokumentieren und stetig überprüfen müssen. Das ist im Einzelfall mit Rechtsunsicherheit verbunden und kann zu Haftungsrisiken führen.

Jegliche genutzte und aufbereitete Datensätze müssen zudem für Verarbeitungsschritte im Zusammenhang mit Datensätzen, die Personenbezug aufweisen (IP-Adressen, Namen, IDs, etc.) zugleich auch datenschutzrechtlichen Vorgaben genügen. Diese Vorgaben gilt es frühzeitig, zum Teil bereits vor Beginn der Entwicklung des KI-Systems, nachweisbar zu erfüllen. Unabhängig davon, ob die Daten vom Anbieter des KI-Systems erhoben oder von einem Dritten oder sogar von der betroffenen Person selbst (öffentlich) bereitgestellt werden, muss die Einhaltung nachgewiesen werden. Werden zudem urheberrechtlich geschützte Inhalte (z. B. Texte, Musik) genutzt, kann es

Quelle: PWC WP Gesellschaft, White Paper Vertrauenswürdige KI, März 2024

# Fallstudie BGB §489 Deployment: EU AI Act

Wir illustrieren einzelne Anforderungen:

**Was davon würde man sowieso tun (müssen)?**

**Was ist Zusatzaufwand?**

**Was davon könnte eine Art „Dealbreaker“ (Deployment Breaker) darstellen?**



Quelle: PWC WP Gesellschaft, White Paper Vertrauenswürdige KI, März 2024

# Fallstudie BGB §489 Deployment: EU AI Act

Wir illustrieren einzelne Anforderungen:

**Was halten Sie von dieser Rollenverteilung?**

**Wie beurteilen Sie die resultierende internationale Wettbewerbsfähigkeit?**

-> Anekdote WLB PC Sicherheit

	1st Line of Defense	2nd Line of Defense	3rd Line of Defense
Zuständigkeit	Data Scientists ML Entwickler MLOps Engineers	Rechtsabteilung Compliance	Interne Revision Externe Prüfung
Aufgaben	Umsetzung auf Use Case Ebene Dokumentation	Rechtliche Vorgaben Aufbau von Kontrollen, Spezifikationen, Standards und Best Practices	Unabhängige Bewertung der Konformität nach Vorgaben des EU AI Acts

können. Denn die Fähigkeit, Qualität, Compliance und Skalierung von KI-Systemen zu verbinden, wird maßgeblich über den Erfolg von Organisationen beim Einsatz von KI auf dem europäischen Markt bestimmen – insbesondere vor dem Hintergrund der technischen Entwicklungen von Wettbewerbern in Asien und den USA. Die Verknüpfung dieser Aspekte erfordert den Einsatz interdisziplinärer Kompetenzen in Organisationen entlang der gesamten KI-Wertschöpfungskette. Nur so lassen sich Strukturen und Prozesse für eine KI-Governance schaffen, die technisch, rechtlich und organisatorisch fit für die Zukunft ist.

Quelle: PWC WP Gesellschaft, White Paper Vertrauenswürdige KI, März 2024

# Fallstudie Firmenkunderating

Wir folgen abermals dem CRISP für diese Fallstudie, bei der es um **finanzielle Fremdschädigung** geht

Ratings sind Schulnoten, um Ausfallwahrscheinlichkeiten auszudrücken

Binär spricht man von Investment Grade (IG) Ratings (gute Kunden) und Sub-IG

Wir wollen prüfen, ob man anhand von Firmenattributen diese beiden Klassen vorhersagen kann

Die Daten und das Beispiel sind von Verbeek (2012, Kap. 7)

Dort kann man auch nachlesen

Wir nutzen [PYTPRA\\_03\\_04\\_LogisticRatingData.ipynb](#)

# Fallstudie Firmenkunderating

- Datensatz von Verbeek (2012, Ch. 7)
- „The files credit contain 921 observations for 2005 for US firms' credit ratings, including a set of firm characteristics. The data are taken from Compustat.“
- Ein RATING < 3 ist spekulativ

**Was ist die beste Ratingstufe?**

**Wieviel Prozent haben ein IG?**

**Mit welcher naiven Vorhersage erhalten Sie eine Genauigkeit (Accuracy) von über 50% OHNE Modellierung?**

	BOOKLEV	EBIT	INVGRADE	LOGSALES	MARKLEV	RATING \
count	921.000000	921.000000	921.000000	921.000000	921.000000	921.000000
mean	0.293187	0.093892	0.472313	7.995754	0.254729	3.499457
std	0.173534	0.084364	0.499504	1.497413	0.187970	1.134561
min	0.000000	-0.384169	0.000000	1.100278	0.000000	1.000000
25%	0.169804	0.051201	0.000000	6.970391	0.109061	3.000000
50%	0.263955	0.090437	0.000000	7.884153	0.211164	3.000000
75%	0.387748	0.135990	1.000000	8.950013	0.347728	4.000000
max	0.999207	0.651508	1.000000	12.701418	0.964859	7.000000

	RETA	WKA
count	921.000000	921.000000
mean	0.156994	0.140414
std	0.300684	0.150340
min	-0.995892	-0.412084
25%	0.011076	0.029047
50%	0.180480	0.122841
75%	0.350836	0.234619
max	0.979922	0.748022

# Fallstudie Firmenkunderating

Wir nutzen `PYTPRA_03_04_LogisticRatingData.ipynb`

Ein RATING < 3 ist spekulativ = Sub-IG

Wir ignorieren RATING im Weiteren und nutzen nur eine der beiden Verschuldungskennzahlen

```
x = data[['BOOKLEV', 'EBIT', 'LOGSALES', 'RETA', 'WKA']]
y = data['INVGRADE']
```

**Was ist zu erklären? Regressand binär?**

**Welche Vorzeichen des Einflusses erwarten Sie?**

Bezeichnung	Erläuterung
Datensatzgröße	<ul style="list-style-type: none"><li>921 Unternehmen</li></ul>
BOOKLEV	<ul style="list-style-type: none"><li>Verschuldung / Buchwert der Assets</li><li>Leverage</li></ul>
EBIT	<ul style="list-style-type: none"><li>Messgröße</li><li>aktuelle Profitabilität</li></ul>
LOGSALES	<ul style="list-style-type: none"><li>Messgröße</li><li>Firmengröße</li></ul>
MARKLEV	<ul style="list-style-type: none"><li>Verschuldung / Marktwert der Assets</li><li>Leverage</li></ul>
RETA	<ul style="list-style-type: none"><li>Einbehaltene Gewinne</li><li>Messgröße</li><li>bisherige Rentabilität</li></ul>
WKA	<ul style="list-style-type: none"><li>Messgröße</li><li>Liquidität auf Basis des WC</li></ul>

# Fallstudie Firmenkunderating

**Interpretieren Sie!**

**Modell verschlanken?**

Optimization terminated successfully.  
Current function value: 0.376108  
Iterations 7

## Logit Regression Results

Dep. Variable:	INVGRADE	No. Observations:	828			
Model:	Logit	Df Residuals:	822			
Method:	MLE	Df Model:	5			
Date:	Wed, 09 Aug 2023	Pseudo R-squ.:	0.4566			
Time:	11:04:44	Log-Likelihood:	-311.42			
converged:	True	LL-Null:	-573.05			
Covariance Type:	nonrobust	LLR p-value:	7.558e-111			
=====						
	coef	std err	z	P> z	[0.025	0.975]
-----						
const	-8.2236	0.915	-8.990	0.000	-10.016	-6.431
BOOKLEV	-4.1983	0.789	-5.322	0.000	-5.744	-2.652
EBIT	4.5684	1.504	3.037	0.002	1.620	7.516
LOGSALES	1.0778	0.101	10.638	0.000	0.879	1.276
RETA	4.1611	0.513	8.118	0.000	3.156	5.166
WKA	-4.0732	0.792	-5.141	0.000	-5.626	-2.520
=====						

# Fallstudie Firmenkunderating

Sei nun  $x_2$  der gesamte Regressorvektor einer Beobachtung  $x_1$  derjenige bei dem der  $k$ -te Regressor um eine Einheit erhöht worden ist.

Konkret seien es hier die Logsales.

Dann ist die dadurch bedingte Veränderung im „Risiko“ wie folgt:

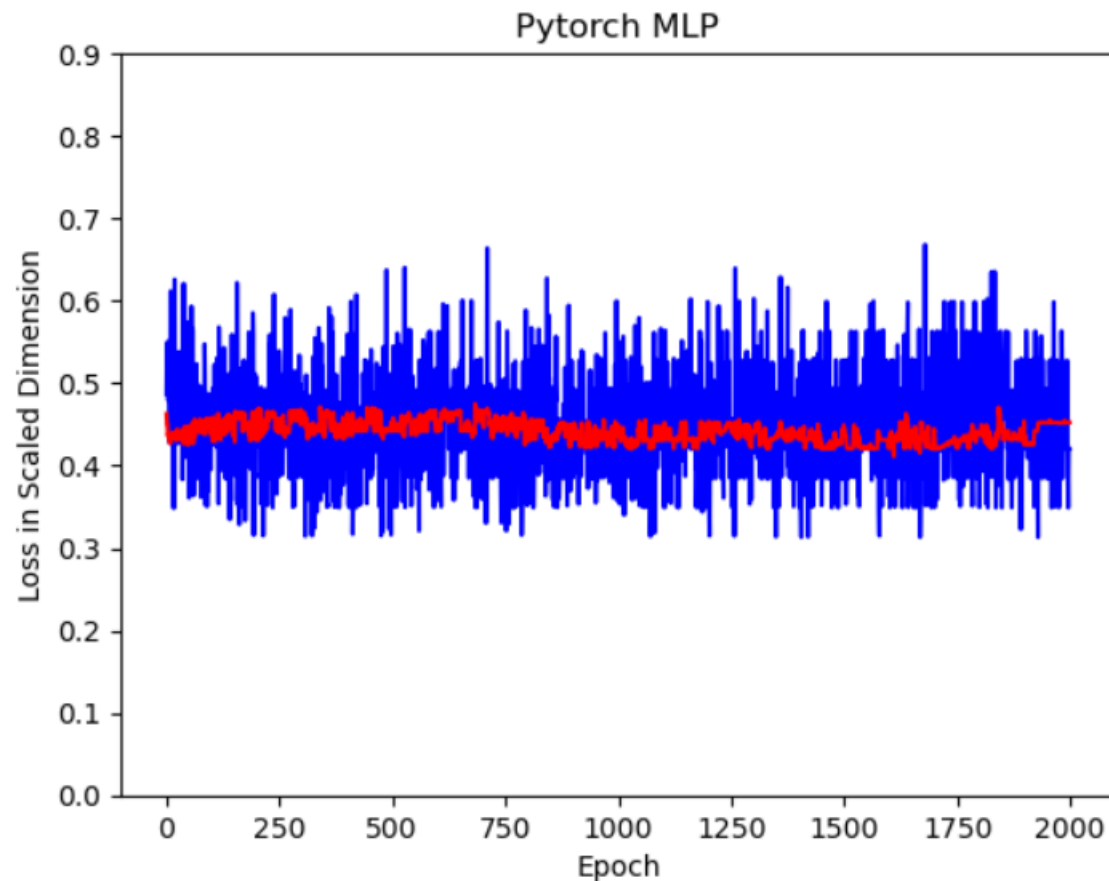
$$\text{Exp}(\beta_k) = \exp(1.0778) = 2.9382 \text{ grob } 3$$

D.h. z. B. die Chance ein IG zu erhalten ist nach Erhöhung der Logsales (+1) approximativ etwa dreimal so gross wie vorher. Diese prominente Rolle ergibt auch eine andere Auswertung.



# Fallstudie Firmenkunderating

Der Trainingsfehler schwankt stärker als der Validierungsfehler, am Ende MLP nicht wirklich besser als Logit



Number of exercises in test data 40  
As percentage 0.430  
Benchmark Accuracy on train data 0.523  
Benchmark Accuracy on test data 0.570

Accuracy Logit on train data: 0.827  
Confusion Matrix Logit on train data:  
[[364 69]  
[ 74 321]]

Accuracy Logit on test data: 0.860  
Confusion Matrix Logit on test data:  
[[43 10]  
[ 3 37]]

Accuracy MLP on train data: 0.867  
Confusion Matrix MLP on train data:  
[[377 56]  
[ 54 341]]

Accuracy MLP on test data: 0.860  
Confusion Matrix MLP on test data:  
[[44 9]  
[ 4 36]]

# Fallstudie Firmenkunderating

Epoche 50 ausreichend

Beide Fehler nahe beieinander

MLP leicht besser, aber auch mehr Paras

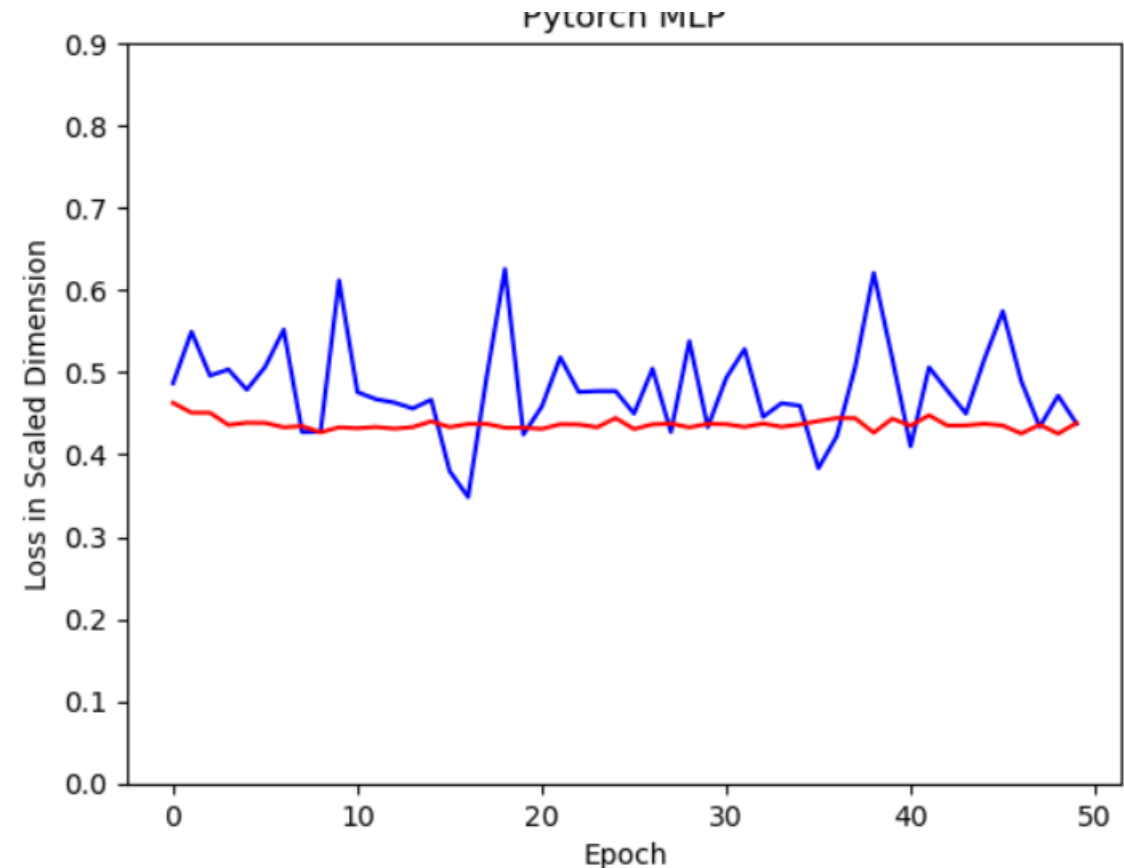
Number of exercises in test data 40  
As percentage 0.430  
Benchmark Accuracy on train data 0.523  
Benchmark Accuracy on test data 0.570

Accuracy Logit on train data: 0.827  
Confusion Matrix Logit on train data:  
[[364 69]  
[ 74 321]]

Accuracy Logit on test data: 0.860  
Confusion Matrix Logit on test data:  
[[43 10]  
[ 3 37]]

Accuracy MLP on train data: 0.864  
Confusion Matrix MLP on train data:  
[[378 55]  
[ 58 337]]

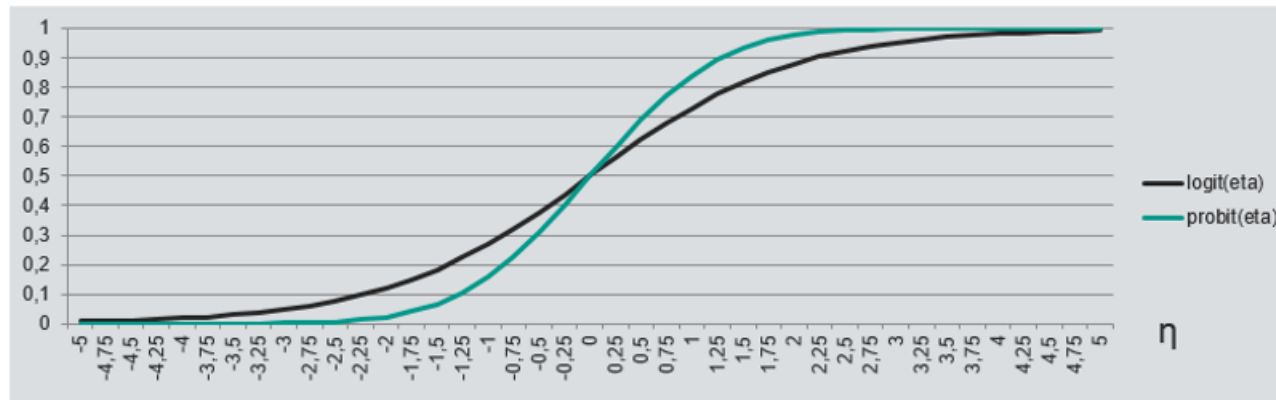
Accuracy MLP on test data: 0.871  
Confusion Matrix MLP on test data:  
[[45 8]  
[ 4 36]]



# Abgrenzung Logit/Probit

## Theorie Nachtrag

- Eine Konsequenz der Modellierung über die latente Variable mit logistischem Störterm lässt sich wie folgt explizit machen. Es sei  $\eta = c + b x_i$ . Dann wissen wir, dass:
  - $P\{y_i = 1 \mid x_i\} = F(\eta)$
- Dabei ist  $F()$  die Verteilungsfunktion der logistischen Verteilung.



- Der Regressor  $X$  wird also zunächst linear transformiert und dann in  $F()$  eingespeist. Damit liegt  $Y$  zwischen 0 und 1.

# Abgrenzung Logit/Probit

## Theorie Nachtrag

- Optisch liegen beide Verteilungsfunktionen nahe beieinander. Deshalb überrascht das Folgende nicht:
- Unterschiede zwischen Probit und Logit
- Gestalt der Verteilung ist leicht unterschiedlich, insbesondere am Rand. Dies merkt man ggf. bei sehr grossen Stichproben, die dann u.U. auch den Rand abdecken
- Die Schätzwerte für die Modellparameter im Logit Modell sind um einen Faktor von ca. 1,8 grösser als im Probit Modell, was an der unterschiedlichen Skalierung der Modelle liegt
- Gemeinsamkeiten zwischen Probit und Logit
- Praktisch führen beide zu ähnlichen Ergebnissen
- Beide werden mit MaxLik geschätzt und beidesmal bewirkt die Konkavität der LogLikelihoodfunction dass es nur ein Maximum gibt

# Abgrenzung Logit/Probit

## Theorie Nachtrag

- Gründe für die Fokussierung auf Logit
- Für das Logit Modell lassen sich viele Berechnungen einfacher durchführen und bestimmte wichtige Eigenschaften exakt zeigen:
- In der Berufserfahrung des Dozenten mit Ratingsystemen spielte nur das Logit Modell eine Rolle.
- Ein wichtiges Thema dabei ist die Unausgewogenheit der Stichprobe. Es gibt oft zu wenige „Ausfälle“. Schröder im Buch „Finanzmarktökonomie“ (2002, 379) weist darauf hin, dass mind. 5% „Ausfälle“ vorhanden sein sollten.
- Problematik wie beim Importance Sampling bei der Monte Carlo Bewertung von Derivaten. Lösungstechnik analog und komfortabel bei Logit Modell.

# Ausblick

- **Ordered Logit**
- **Multinomial Logit:** Individuum wählt zwischen  $a$  Alternativen, wobei keine Ordnung zwischen  $A_i$  und  $A_j$  a priori herstellbar (Marketing Ökonometrie, s.u.)
- Weitere Beispiele:
  - Kreditrückzahlung: Kunde kündigt Kredit vorzeitig oder Kunde zahlt Kredit wie vorgesehen ab oder Kunde kann Kredit nicht abbezahlen
  - Spezialfall Kreditausfall: ja oder nein?
  - Börsengang: ja oder nein?
  - Feindliche Übernahme: ja oder nein?
  - Dividendenausschüttung: ja oder nein?
- Aspekte bei Multinomial Logit:
  - Conjoint Analyse: Regressoren erklären eine Rangordnung
  - Conditional Logit: Produktmerkmale als Regressoren
  - Mixed Logit: Produkt- und Kundenmerkmale als Regressoren

**Welche Anwendungen kommen Ihnen für Ihre Praxis in den Sinn?**

# Fehlerquellen

Ergänzend noch ein paar Hinweise auf mögliche **technische** Fehlerquellen.

Wir besprechen nachfolgend Auszüge von [https://uvadlc-notebooks.readthedocs.io/en/latest/tutorial\\_notebooks/guide3/Debugging\\_PyTorch.html](https://uvadlc-notebooks.readthedocs.io/en/latest/tutorial_notebooks/guide3/Debugging_PyTorch.html) und [https://www.learnpytorch.io/pytorch\\_most\\_common\\_errors/](https://www.learnpytorch.io/pytorch_most_common_errors/), Zugriff 13. Juli 2023

## Classics

1. Shape errors - You're trying to perform an operation on matrices/tensors with shapes that don't line up. For example, your data's shape is [1, 28, 28] but your first layer takes an input of [10].
2. Device errors - Your model is on a different device to your data. For example your model is on the GPU (e.g. "cuda") and your data is on the CPU (e.g. "cpu").
3. Datatype errors - Your data is one datatype (e.g. torch.float32), however the operation you're trying to perform requires another datatype (e.g. torch.int64).

# Fehlerquellen

## Categorical Forecasting

The most common mistake is the mismatch between loss function and output activation function. The loss module `nn.CrossEntropyLoss` in PyTorch performs two operations: `nn.LogSoftmax` and `nn.NLLLoss`. Hence, the input to this loss module should be the output of your last linear layer. Do not apply a softmax before the Cross-Entropy loss. Otherwise, PyTorch will apply a log-softmax on your softmax outputs, which will significantly worsen the performance, and give you headaches.

If you use the loss module `nn.NLLLoss`, you need to apply the log-softmax yourself. `NLLLoss` requires log-probabilities, not plain probabilities. Hence, make sure to apply `nn.LogSoftmax` or `nn.functional.log_softmax`, and not `nn.Softmax`.

Softmax over the right dimension

Pay attention to the dimension you apply your softmax over. Usually, this is the last dimension of your output tensor, which you can identify with e.g. `nn.Softmax(dim=-1)`. If you mix up the dimension, your model ends up with random predictions.

**Diese Ausführungen werden wir erst nach den nächsten Beispielen zuordnen können!**



# Fehlerquellen

## Initialization

Initializing the parameters of your model correctly is very important (see Tutorial 4 for details on this). Initializing parameters with a standard normal distribution is not a good practice and often fails. It can occasionally work for very shallow networks, but don't risk it! Think about your initialization, and use proper methods like Kaiming or Xavier.

## Zero-grad in optimizers

Remember to call `optimizer.zero_grad()` before doing `loss.backward()`. If you do not reset the gradients for all parameters before performing backpropagation, your gradients will be added to those from the previous batch. Hence, your gradients end up to be not the ones you intended them to be.

## Weight decay and Adam

Adam is known to have a different implementation of weight decay in many frameworks than you would expect. Specifically, the weight decay is usually added as gradients before determining the adaptive learning rate, and hence scaling up the weight decay for parameters with low gradient norms. Details on this problem, which is actually shared across most common DL frameworks, can be found [here](#). In PyTorch, you can use the desired version of weight decay in Adam using `torch.optim.AdamW` (identical to `torch.optim.Adam` besides the weight decay implementation).

# Rückblick

- Wir haben den Übergang von klass. Modellierung mit vorheriger Spezifizierung zu moderner datengetriebener Modellierung erlebt
- Vorher:  $y=a+bx$  als Spezifikation der Schätzgleichung.

## Welche Spezifizierung haben Sie noch kennengelernt?

- Jetzt: Während Training wird spezifiziert.

## Welche Theoreme lassen uns da hoffnungsvoll sein?

- Es gibt noch mehr. MAKI