DAV
Deutsche
Aktuarvereinigung e.V.

Schriftliche Prüfung im Fach

Actuarial Data Science Advanced

gemäß Prüfungsordnung 5
der Deutschen Aktuarvereinigung e. V.

am 24. Oktober 2025

Hinweise:

- Als Hilfsmittel ist ein Taschenrechner zugelassen.
- Die Gesamtpunktzahl beträgt 180 Punkte. Die Klausur ist bestanden, wenn mindestens 90 Punkte erreicht werden.
- Bitte prüfen Sie die Ihnen vorliegende Prüfungsklausur auf Vollständigkeit. Die Klausur besteht aus 36 Seiten.
- Alle Antworten sind zu begründen und bei Rechenaufgaben muss der Lösungsweg ersichtlich sein.
- Bitte vermeiden Sie bei der Lösungserstellung die nicht zusammenhängende Streuung der Lösungen zu den einzelnen Aufgabenteilen.
- Aus Gründen der besseren Lesbarkeit wird auf die gleichzeitige Verwendung der Sprachformen männlich, weiblich und divers (m/w/d) verzichtet.

Mitglieder der Prüfungskommission:

Axel Kiermaier, Dr. René Külheim, Prof. Dr. Jonas Offtermatt

Aufgabe 1. *[1.1 Gesellschaftliches Umfeld & Ethik 2, 3.4 Innovative Produkte] [22 Punkte]*

Sie sind als Data Scientist und Aktuar bei der **SecureEverything AG** tätig. Ihr Unternehmen steht, wie jedes Unternehmen aktuell, vor der Herausforderung, neue regulatorische Vorgaben zu erfüllen, mit disruptiven Marktveränderungen umzugehen und innovative Produkte zu entwickeln, um wettbewerbsfähig zu bleiben. Der Vorstand hat Sie gebeten, eine Analyse zu erstellen und konkrete Empfehlungen für die Unternehmensstrategie auszuarbeiten.

Hinweis: Ihr Auftraggeber ist Ihr Vorstand. Schreiben Sie also Management Summaries und keine wissenschaftlichen Ausarbeitungen.

(a) *[6 Punkte]* Der Vorstand möchte wissen, wie KI-Systeme innerhalb des Artifical Intelligence Acts der EU klassifiziert werden und Ihre begründete(!) Einschätzung erfahren, in welche Kategorie die automatisierte Schadenregulierung mittels KI fällt.

(b) *[4 Punkte]* Der Vorstand ist besorgt über mögliche Disruptionsgefahren, die durch technologische Entwicklungen und neue Marktteilnehmer entstehen. Nennen und erläutern Sie zwei konkrete Disruptionsgefahren, die für traditionelle Versicherer wie die SecureEverything AG entstehen können.

(c) *[6 Punkte]* Um mit dem Wettbewerb mithalten zu können, sollen mittels Data Science neue innovative Produkte entwickelt werden. Benennen Sie drei Vorteile, welche Data Science Produkte für Versicherer oder Kunden haben können.

(d) *[6 Punkte]* Nachdem Sie dem Vorstand die drei Handlungsfelder (vermehrte Regulatorik, Disruptionsgefahren, neue innovative Produkte am Markt) verständlich gemacht haben, fordert er Sie auf, Handlungsempfehlungen für die SecureEverything AG vorzuschlagen. Nennen Sie kurz und knapp drei konkrete Empfehlungen, basierend auf den Handlungsfeldern, denen das Unternehmen Ihrer Meinung nach in der nahen Zukunft folgen sollte.

Lösungsvorschlag:

(a) Der Artificial Intelligence Act (AI Act) der EU klassifiziert KI-Systeme in vier Risikostufen:

- Unzulässiges Risiko (z. B. Social Scoring)
- Hohes Risiko (z. B. Kreditvergabe)
- Begrenztes und minimales Risiko (z. B. Chatbots, Empfehlungssysteme)
- Kein Risiko

Die automatisierte Schadenregulierung mittels KI fällt voraussichtlich in die Kategorie „hohes Risiko“, da sie in den sensiblen Bereich der Finanzdienstleistungen fällt und erhebliche Auswirkungen auf Verbraucher haben kann. Eine strikte Compliance mit Transparenz- und Fairness-Anforderungen ist notwendig.

(1 Punkte pro Risikoklasse, 2 für die Begründung zur Schadensregulierung)

(b) Zwei Disruptionsgefahren wären beispielsweise:

- InsurTech-Startups & BigTech-Einstieg: Technologiegetriebene Startups und große Technologieunternehmen (z. B. Google, Amazon) könnten mit datenbasierten Geschäftsmodellen in den Versicherungsmarkt eindringen.
- Pay-as-you-go- & On-Demand-Versicherungen: Neue flexible Modelle basierend auf Echtzeit-Daten (Telematik, IoT) bedrohen klassische Policen. Kunden wollen situative, personalisierte Policen statt langfristiger Verträge.

(2 Punkte pro Disruptionsgefahr, andere Nennungen sind möglich)

(c) Vorteile und Möglichkeiten der Anwendung von Data Science:

- Aufbau von detaillierten Datengrundlagen zur Risikobewertung
- Genauere Differenzierung und Selektion von Risiken
- Verbesserte Möglichkeit zur versicherungstechnischen Bewertung und Analyse von Risiken (u.a. Betrugserkennung und Schadenvermeidung)
- Aufbau von detaillierten Datengrundlagen zur Bewertung der Kundenbeziehung
- Verbesserte Analyse und Modellierung der Kundenbeziehung
- Optimierung von Kundengruppenmanagement und Database Marketing
- …

(Pro genannten Vorteil 2 Punkte bis zu maximal 6 Punkten, andere Nennungen sind möglich)

(d) Hier ist Kreativität und Management-Denken gefragt. Es gibt somit auch kein konkretes richtig oder falsch. Mögliche Nennungen wären:

- Regulatorische Compliance strategisch integrieren: Frühzeitige Anpassung an den AI Act und DSGVO-Richtlinien, um Risiken zu vermeiden. Installation eines AI Officers?
- Fokus auf datengetriebene Innovationen: Entwicklung personalisierter Tarife & KI-gestützter Risikomodelle zur Differenzierung am Markt. Einsatz eines aktuariellen Data-Science-Teams?
- Kooperation mit InsurTechs & Technologiefirmen: Partnerschaften mit Startups oder Tech-Konzernen, um Innovationskraft und digitale Fähigkeiten zu stärken. Teilnahme an Innovationsnetzwerken?

(Pro Handlungsempfehlung 2 Punkte)

Aufgabe 2. *[3.3 Visualisierung 1] [20 Punkte]*

Die **SecureEverything AG** analysiert Schadensfälle in der Kfz-Versicherung, um Tarife zu optimieren. Sie erhalten eine Stichprobe von **10 Schadensfällen** mit den Merkmalen **Fahreralter, Fahrzeugtyp, Schadenshöhe** und dem **Kilometerstand**.

Fahreralter	Fahrzeugtyp	Schadenshöhe	Kilometerstand
22	Kleinwagen	0	50.000
45	SUV	5.000	NaN
33	Kombi	0	80.000
60	SUV	7.500	120.000
29	Kleinwagen	0	30.000
50	Kleinwagen	1.300	20.000
38	Kombi	3.200	75.000
27	SUV	6.000	100.000
41	Kleinwagen	1.800	45.000
55	Kombi	2.200	NaN

Erstellen Sie für die Beantwortung der folgenden drei Fragen jeweils **eine geeignete grafische Darstellung** (eine Interpretation der Grafiken ist nicht gefordert):

(a) [8 Punkte] Gibt es einen Zusammenhang zwischen Fahrzeugtyp und der Wahrscheinlichkeit, dass ein Schaden vorliegt?

(b) [7 Punkte] Welchen Einfluss hat der Fahrzeugtyp auf die Schadenshöhe?

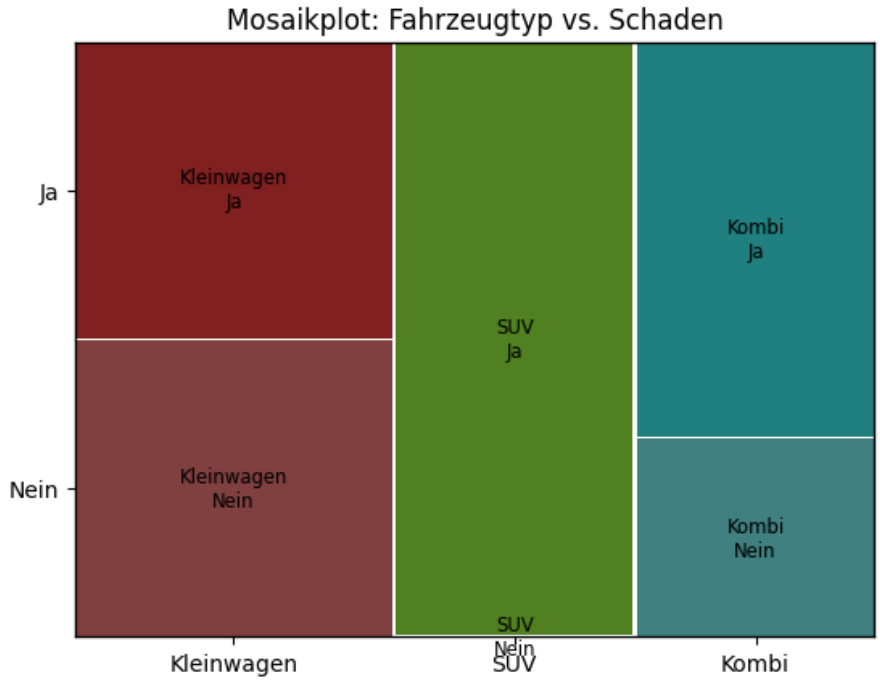
(c) [5 Punkte] Wie ist die Verteilung von Fällen mit Schaden und ohne Schaden?

Hinweis:
Erstellen Sie **mindestens einen Mosaikplot und einen vereinfachten Boxplot (ohne Quantilswerte)**, um die Fragen zu beantworten. Wählen Sie darüber hinaus weitere geeignete Diagrammtypen zur Beantwortung der Fragen. Es ist keine Begründung verlangt, lediglich geeignete Schaubilder.

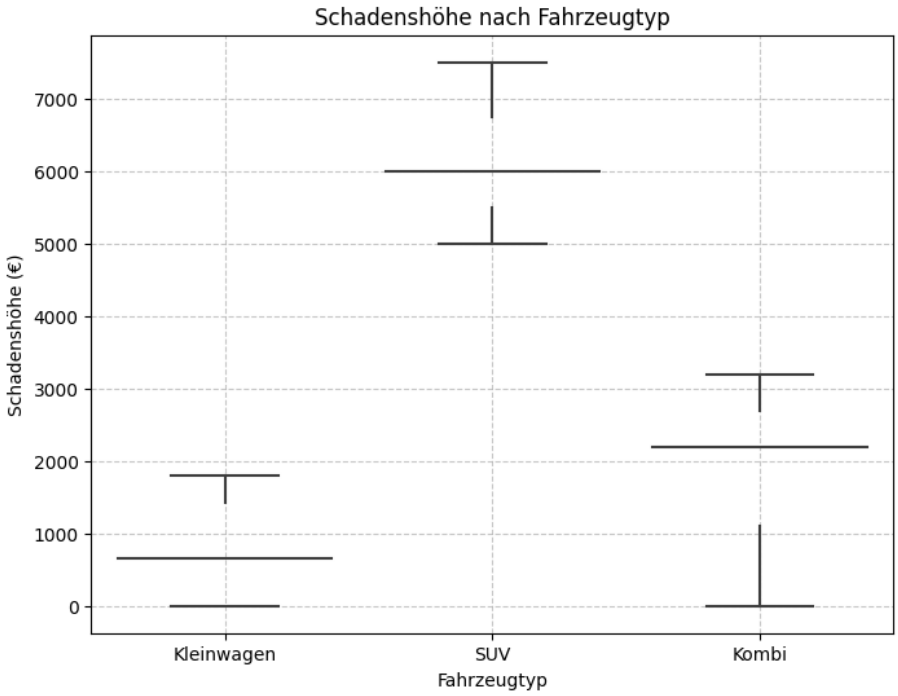
Bitte denken Sie an die grundlegenden Anforderungen an Visualisierungen: Lineal verwenden, Achsen beschriften, Legenden, Titel, etc.

Lösungsvorschlag:

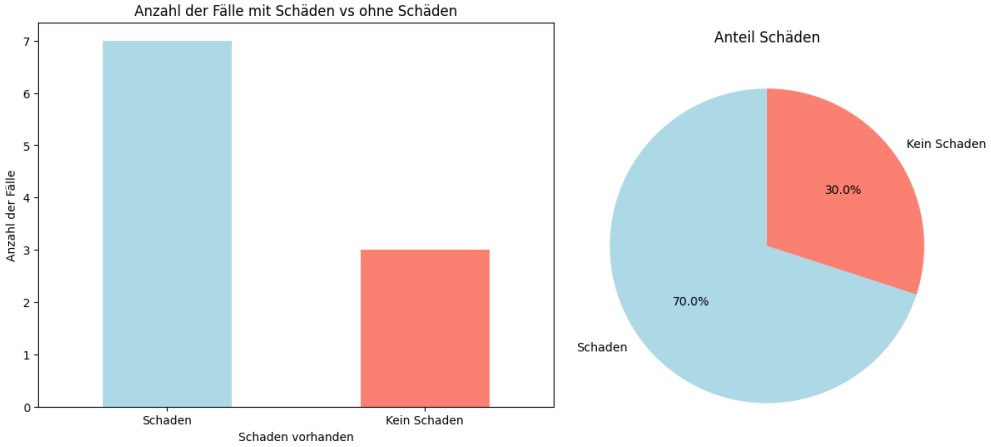
(a) Mögliche Lösung ein mosaicplot:



(b) Mögliche Lösung ein vereinfachter boxplot:



(c) Mögliche Lösung ein Balken- oder Kreisdiagramm:



Aufgabe 3. *[4.4 Datenaufbereitung zur Modellerstellung, 5.1 Programmiersprachen für Data Science] [20 Punkte]*

In dieser Aufgabe arbeiten Sie erneut mit dem Datensatz von Schadensfällen einer KFZ-Versicherung aus Aufgabe 2. Gehen Sie davon aus, dass die Daten mittels folgenden Codes in einen Pandas-Dataframe überführt wurden:

```
import pandas as pd

# DataFrame erstellen

data = {

    'Fahreralter': [22, 45, 33, 60, 29, 50, 38, 27, 41, 55],

    'Fahrzeugtyp': ['Kleinwagen', 'SUV', 'Kombi', 'SUV', 'Kleinwagen',
'Kleinwagen', 'Kombi', 'SUV', 'Kleinwagen', 'Kombi'],

     'Schadenshöhe': [0, 5000, 0, 7500, 0, 1300, 3200, 6000, 1800,
2200],

    'Kilometerstand': [50000, None, 80000, 120000, 30000, 20000, 75000,
100000, 45000, None]

}

df = pd.DataFrame(data)
```

Bitte geben Sie jeweils die Ausgaben des print-Befehls der unten stehenden Python-Code-Zeilen an.

(a) *[5 Punkte]*

```
suv_df=
df[df['Fahrzeugtyp']=='SUV'].reset_index(drop=True)

print(suv_df)
```

(b) *[5 Punkte]*

```
age_over_30 = df[df['Fahreralter'] < 30]

print(age_over_30[['Fahreralter', 'Fahrzeugtyp']])
```

(c) *[5 Punkte]*

```
df['Schadenquote']=df['Schadenshöhe'] / df['Kilometerstand']

print(df.iloc[7])
```

(d) *[5 Punkte]*

```
df['Kilometerstand']=df['Kilometerstand'].fillna(df['Kilometer
stand'].median())

print(df['Kilometerstand'])
```

Lösungsvorschlag:

(a) Achtung, hier muss der Index korrekt angegeben werden. Dieser wird neu erzeugt.

	Fahreralter	Fahrzeugtyp	Schadenshöhe	Kilometerstand
0	45	SUV	5000	NaN
1	60	SUV	7500	120000.0
2	27	SUV	6000	100000.0

(b) Achtung, Variablenname passt nicht zum „<“-Operator.

	Fahreralter	Fahrzeugtyp
0	22	Kleinwagen
4	29	Kleinwagen
7	27	SUV

(c) Nur die achte Zeile der Tabelle (Python startet bei 0).

```
Fahreralter             27
Fahrzeugtyp            SUV
Schadenshöhe          6000
Kilometerstand    100000.0
Schadenquote          0.06
Name: 7, dtype: object
```
(Diese Zeile muss für die volle Punktzahl nicht angegeben werden)

(d) Es muss die ganze Spalte ausgegeben werden, nicht nur die imputierten Werte. Die letzte Zeile und der Index müssen für die volle Punktzahl nicht angegeben werden.

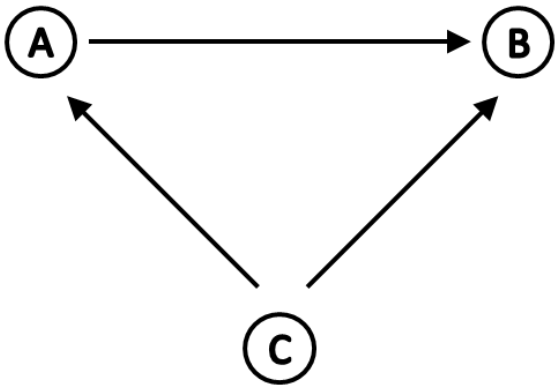
```
0     50000.0
1     62500.0
2     80000.0
3    120000.0
4     30000.0
5     20000.0
6     75000.0
7    100000.0
8     45000.0
9     62500.0
Name: Kilometerstand, dtype: float64
```
(Diese Zeile muss für die volle Punktzahl nicht angegeben werden)

Aufgabe 4. *[4.3 Korrelation & kausale Inferenz] [19 Punkte]*

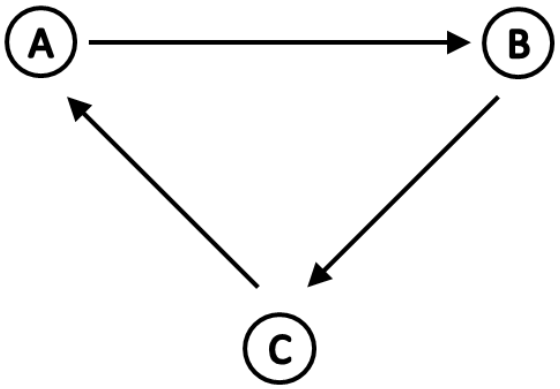
(a) *[12 Punkte]* Erläutern Sie die Konzepte Survival Bias, Outcome Bias und Omitted Variable Bias und nennen Sie jeweils ein Beispiel aus dem Versicherungsumfeld. Beantworten Sie ebenfalls für jedes dieser Konzepte einzeln, was es mit dem Thema Kausalität zu tun hat.

(b) *[7 Punkte]* Bei den folgenden vier Graphen (b1) – (b4) steht die Kausalbeziehung zwischen *A* und *B* im Mittelpunkt. Welche Rolle spielt in diesem Zusammenhang jeweils *C*? Bestimmen Sie jeweils die gemeinsame Wahrscheinlichkeitsverteilung *P(A,B,C)*.

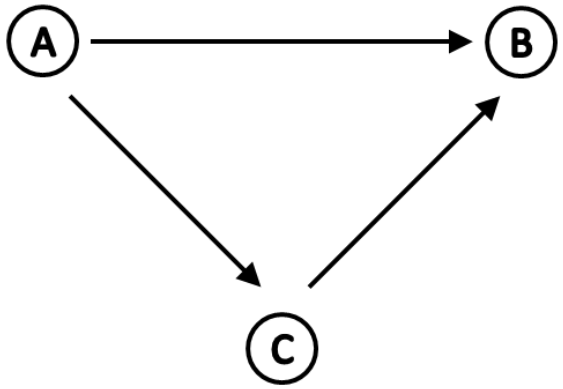
(b1):



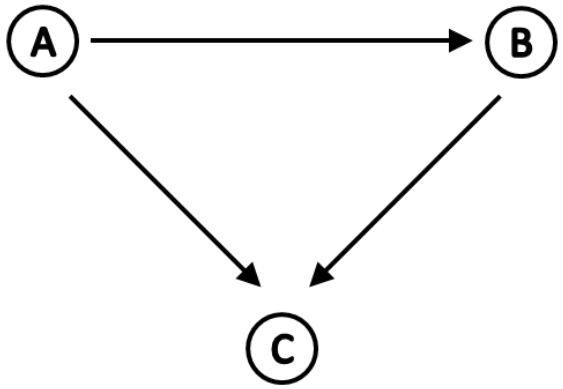
(b2):



(b3):



(b4):



Lösungsvorschlag:

(a) Survival Bias, Outcome Bias und Omitted Variable Bias sind verschiedene Formen von statistischen Verzerrungseffekten.

1. Survival Bias

Der Survival Bias oder die Überlebenden-Verzerrung besteht darin, die Wahrscheinlichkeit des Überlebens bzw. eines Erfolgs systematisch zu überschätzen, weil Überlebende / Erfolgreiche mit höherer Wahrscheinlichkeit in einer Statistik auftauchen als Erfolglose.

Beispiel: Analyse der Zufriedenheit der Teilnehmer an einem bereits seit langem laufenden Bonusprogramm. Hier haben die meisten unzufriedenen Teilnehmer bereits in der Vergangenheit gekündigt, wodurch der Anteil der zufriedenen Teilnehmer an den „lebenden“ Teilnehmern überhöht ist.

Zusammenhang zu Kausalität: Beim Survival Bias werden nur selektive und somit verzerrte Stichproben betrachtet. Damit können bei bestimmten Analysen die Zusammenhänge und wahren Ursachen nicht erkannt werden.

2. Outcome Bias

Der Outcome Bias oder die Ergebnisverzerrung entsteht dadurch, dass die Qualität einer Entscheidung anhand ihres Ergebnisses beurteilt wird. Bei einem positiven Ergebnis wird damit im Nachhinein die Entscheidung als gut eingestuft, ggf. gemäß dem Prinzip „das Ergebnis rechtfertigt die Mittel“; mögliche Risiken werden damit ignoriert. Bei einem negativen Ergebnis wird dagegen im Nachhinein die Entscheidung als schlecht eingestuft, obwohl sie auf Basis der zum Zeitpunkt der Entscheidung vorliegenden Informationen die bestmögliche war.

Beispiel: Die Risikoprüfung eines Antrags führte zu dessen Annahme; später tritt ein hoher Schaden bei dem Vertrag ein. Eine nachträgliche Beurteilung des Risikoprüfers anhand des Ergebnisses ist nicht unbedingt adäquat.

Zusammenhang zu Kausalität: Beim Outcome-Bias wird unterstellt, dass zwischen Entscheidung und Ergebnis ein Kausalzusammenhang besteht; dabei werden gegebenenfalls vorhandene Unsicherheiten zum Zeitpunkt der Entscheidung außer Acht gelassen. Daraus ergibt sich, dass mögliche Risiken der Entscheidung ignoriert werden. Damit wird z.B. eine Entscheidungsfindung als positiv dargestellt, obwohl das positive Ergebnis nicht als Kausalwirkung, sondern lediglich als Folge glücklicher Umstände eingetreten ist.

3. Omitted Variable Bias

Der Omitted Variable Bias oder die Verzerrung durch ausgelassene Variablen liegt vor, wenn eine oder mehrere relevante Variablen außer Acht gelassen werden.

Beispiel: Ein Unternehmen stellt höhere Unfallhäufigkeiten bei roten Fahrzeugen fest und berücksichtigt deshalb die Farbe als Risikofaktor, nicht aber das Fahrverhalten (wozu keine Daten vorliegen). Tatsächlich kann eine gewisse Farbpräferenz mit einem bestimmten Fahrverhalten korrelieren, ist aber nicht ursächlich dafür und damit auch nicht ursächlich für eine höhere Unfallhäufigkeit.

Zusammenhang zu Kausalität: In einem DAG nicht berücksichtigte Confounder (wie im Beispiel mangels Telematik fehlende Informationen zum Fahrverhalten) führen zu haltlosen Kausalaussagen.

(b) In **(b1)** ist *C* ein Confounder (Störgröße) für die Wirkung von *A* auf *B*. Er beeinflusst sowohl *A* wie auch *B*. Ein Confounder kann zu falschen Schlussfolgerungen führen, wenn er nicht angemessen kontrolliert oder berücksichtigt wird.

$$P(A,B,C) = P(B|A,C)\,P(A,C) = P(B|A,C)\,P(A|C)\,P(C)$$

In **(b2)** liegt kein DAG vor. Der Graph ist zyklisch und somit nicht für die Abbildung eines Kausalzusammenhangs geeignet.

Insbesondere lässt sich $P(A,B,C)$ nicht in ähnlicher Weise faktorisieren.

In **(b3)** erfolgt die Wirkung von *A* auf *B* nicht (nur) direkt, sondern teilweise über *C*. *(Anm.: C wird hier auch als Mediator bezeichnet.)* Zur Bestimmung des direkten Effekts von *A* auf *B* könnte man durch geeignete Verfahren den totalen Effekt von *A* auf *B* (z.B. durch Regression) sowie den Mediationseffekt von *C* (z.B. durch eine Pfadanalyse) getrennt bestimmen und die Differenz hieraus bilden.

$$P(A,B,C) = P(B|A,C)\,P(A,C) = P(B|A,C)\,P(C|A)\,P(A)$$

In **(b4)** ist *C* ein Collider. Gegeben *C* werden *A* und *B* (auch wenn sie unabhängig sind) bedingt abhängig. Hierdurch können kontraintuitive Korrelationen entstehen.

$$P(A,B,C) = P(C|A,B)\,P(A,B) = P(C|A,B)\,P(B|A)\,P(A)$$

Aufgabe 5. *[2.1 Datenmanagement 2 – Relationale DBs] [26 Punkte]*

Vorbemerkung: Bitte nutzen Sie in dieser Aufgabe für ER-Diagramme durchgängig die Martin-Notation (Krähenfußnotation).

(a) *[6 Punkte]* Definieren Sie die erste bis dritte Normalform und erläutern Sie die Anforderungen dieser Definitionen jeweils anhand eines minimalen Beispiels aus dem Versicherungskontext, in dem die jeweilige Anforderung verletzt ist.

(b) *[12 Punkte]* Entwickeln Sie ein ER-Diagramm aus folgender Beschreibung einer Versicherungssituation aus dem Bereich der Lebensversicherung. Gehen Sie auf die Entitäten (Objekttypen), Attribute, Beziehungen und Kardinalitäten ein und zeichnen Sie dann das Diagramm.

Im Zentrum Ihrer Versicherung soll die natürliche Person stehen. Sie wird beschrieben durch Vorname, Nachname, Geburtsdatum. Personen können miteinander verheiratet sein, was für eines Ihrer Hauptprodukte (Versicherung auf verbundene Leben) von besonderer Bedeutung ist.

Zur Beschreibung eines Versicherungsvertrags beschränken Sie sich in diesem vereinfachten Beispiel auf Versicherungssumme, ratierlicher Beitrag, Zahlweise, Beginn- und Ablaufdatum.

Personen können im Rahmen eines Versicherungsvertrags verschiedene Rollen einnehmen: versicherte Person, Versicherungsnehmer, Beitragszahler.

Personen haben eine Adresse (Straße, Hausnummer, PLZ, Ort).

Modellieren Sie die Rolle und die Adresse als eigenen Objekttyp.

(c) *[3 Punkte]* Nennen Sie je zwei Vor- und Nachteile der in (b) vorgenommenen expliziten Modellierung der Rolle als Objekttyp. Welchen Vorteil würden Sie besonders hervorheben?

(d) *[5 Punkte]* "Übersetzen" Sie das Beispiel aus Teilaufgabe (b) in Tabellen-Typbeschreibungen.

Lösungsvorschlag:

(a) 1. Normalform:

Ein Datenmodell liegt in der ersten Normalform vor, wenn es keine multiplen Eigenschaften aufweist, d.h. alle Attribute atomar sind.

Gegenbeispiel:

Eine Tabelle Person enthalte die Attribute Vorname, Nachname und Telefonnummern. Dabei werden in letzterem Attribut mehrere Telefonnummern als kommaseparierter String gespeichert, also etwa

Person_ID	Vorname	Nachname	Telefonnummern
1	Peter	Meier	010/1234567, 089/9876543

Hier ist das Attribut Telefonnummern nicht atomar und daher ist die erste Normalform verletzt.

2. Normalform:

Ein Datenmodell liegt in der zweiten Normalform vor, wenn es sich in der ersten Normalform befindet und jede beschreibende Eigenschaft eines Objekttyps zwar vom Gesamtschlüssel, aber nicht bereits von einem Teilschlüssel dieses Objekttyps funktional abhängig ist.

Gegenbeispiel:

Eine Tabelle Vertragsbeziehung enthalte die Attribute Vertrag_ID und Person_ID (die gemeinsam den Primärschlüssel der Tabelle bilden) und Beginndatum, also etwa:

Vertrag_ID	Person_ID	Beginndatum
1234567	0000001	01.10.2025
1234567	0000002	01.10.2025

Hier hängt das Attribut Beginndatum bereits eindeutig von der Vertrag_ID ab, die aber nur ein Teilschlüssel und nicht der Gesamtschlüssel der Tabelle ist. Daher ist die zweite Normalform verletzt.

3. Normalform:

Ein Datenmodell liegt in der dritten Normalform vor, wenn es sich in der zweiten Normalform befindet und keine beschreibende Eigenschaft eines Objekttyps von einer anderen beschreibenden Eigenschaft dieses Objekttyps funktional abhängig ist.

Gegenbeispiel:

Eine Tabelle Vertrag enthält die Attribute Versicherungssumme, Produktschlüssel und Produktname, also etwa

Vertrag_ID	Versicherungssumme	Produktschlüssel	Produktname
1234567	500.000	P001	LV_Klassik
1234568	100.000	P001	LV_Klassik
1234569	20.000	P002	LV_Fonds
1234570	30.000	P002	LV_Fonds

Hier hängt der Wert des Attributs Produktname funktional unmittelbar von dem Attribut Produktschlüssel ab. Daher ist die dritte Normalform verletzt.

(b) Aus der Beschreibung ergeben sich folgende Entitäten und Attribute (hier mit englischen Bezeichnern – deutsche Bezeichner sind ebenso möglich):

1. Entität „person“

   Attribute:

   - firstName
   - lastName
   - dateOfBirth

2. Entität „address“

Attribute:

- street
- houseNumber
- zipCode
- city

3. Entität „contract“

   Attribute:

   - insuranceAmount
   - periodicContribution
   - paymentMethod
   - startDate
   - expiryDate

4. Entität „role“

   Attribute:

   - roleDescription

Beschreibung der Beziehungen und Kardinalitäten:

1. address zu person

   Eine Person hat genau eine Adresse.

   Eine Adresse kann für eine oder mehrere Personen gelten (z.B. mehrere Personen in einem Haushalt).

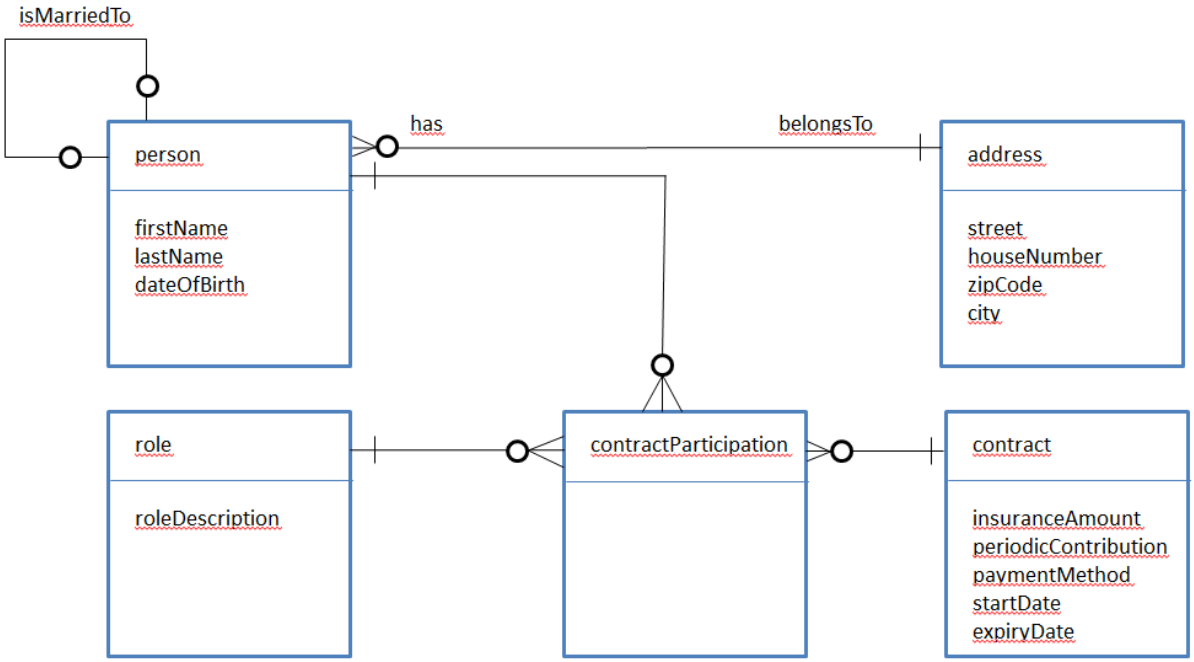
2. person zu person (rekursive Beziehung)

   Jede Person ist mit keiner oder genau einer Person verheiratet.

3. contractParticipation (assoziative Entität)

Zur Abbildung des Umstandes, dass eine Person mehrere verschiedene Rollen in einem Versicherungsvertrag einnehmen kann, wird für die Vertragsbeteiligung eine assoziative Entität eingeführt.

Hieraus ergibt sich folgendes ER-Diagramm:



(c) Vorteile:

- Normalisierung
- Flexibilität / Erweiterbarkeit

Nachteile:

- Performance-Effekte
- Komplexität / Überdimensionierung

Ein besonders wichtiger Vorteil der expliziten Modellierung der Rolle ist die Flexibilität für künftige Erweiterungen. Wenn also z.B. davon auszugehen ist, dass in Zukunft weitere Rollen hinzukommen, zeigt diese Form der Modellierung einen unschätzbaren Vorteil.

(d) Das Beispiel aus Teilaufgabe (b) lässt sich folgendermaßen in Tabellen-Typbeschreibungen abbilden:

person (**personID**, firstName, lastName, dateOfBirth, ⇑addressID⇑, *⇑isMarriedTo-personID⇑*)

address (**addressID**, street, houseNumber, zipCode, city)

contract (**contractID**, insuranceAmount, periodicContribution, paymentMethod, startDate, expiryDate)

role (**roleID**, roleDescription)

contractParticipation (⇑**personID**⇑ + ⇑**contractID**⇑ + ⇑**roleID**⇑)

Aufgabe 6. *[3.1 Data Mining 2] [23 Punkte]*

(a) [5 Punkte] Erläutern Sie in einem Satz, was CRISP-DM ist. Wofür steht die Abkürzung? Nennen Sie die einzelnen Phasen von CRISP-DM.

(b) [18 Punkte] Nennen Sie für jede Phase zwei Aufgaben in der Terminologie von CRISP-DM und erläutern Sie diese Aufgaben anhand eines Beispiels aus dem Versicherungsumfeld. Gehen Sie dabei durchgängig von einem Projekt zur Stornoverhütung aus.

Lösungsvorschlag:

(a) CRISP-DM = Cross Industry Standard Process for Data Mining.

CRISP-DM ist ein Prozessmodell für Data-Mining-Projekte, das 1999/2000 von einem Konsortium von Firmen aus mehreren Branchen erstellt wurde und das auch heute noch aufgrund seiner Verbreitung als das Standardmodell angesehen wird.

Die 6 Phasen von CRISP-DM heißen

- Geschäftsverständnis
- Datenverständnis
- Datenvorbereitung
- Modellierung
- Evaluierung
- Bereitstellung / Anwendung

(b) *Vorbemerkung: Die folgende Lösung ist nur beispielhaft zu verstehen. Andere Antworten sind möglich. Maßgeblich für das Erreichen der vollen Punktzahl ist das korrekte Benennen und Zuordnen von je zwei Aufgaben zu den einzelnen Phasen gemäß CRISP-DM sowie die Beschreibung eines möglichst stornospezifischen Beispiels.*

- Phase Geschäftsverständnis
  - Aufgabe „Bestimmen von Geschäftszielen“

    Beispiel:

    Das Unternehmen möchte seine Bestandsgröße sichern, indem die Stornoquote reduziert wird.

  - Aufgabe „Bestimmen von Data-Mining-Zielen“

    Beispiel:

    Um gezielte Maßnahmen zur Verbesserung der Kundenbindung einzuleiten, werden entsprechende Ziele für das Data Mining festgelegt. Zum Beispiel soll eine korrekte Vorhersage eines gewissen Mindestprozentsatzes an stornogefährdeten Kunden erreicht werden.

- Phase Datenverständnis

- Aufgabe „Sammeln ursprünglicher Daten“

  Beispiel:

  Als Datenquellen kommen hier insbesondere die Vertragsdatenbank(en) und die Kundeninteraktionen (Korrespondenz, Anrufe) in Frage, aber auch Informationen zum Marktumfeld (wie zum Beispiel zur Zinssituation).

- Aufgabe „Überprüfen der Datenqualität“

  Beispiel:

  Wichtig ist z.B. die Vollständigkeit der Daten. Wenn etwa aus historischen Gründen die gewählten Datenquellen nicht alle Verträge enthalten, könnten hierdurch Verträge mit vom Rest abweichendem Stornoverhalten außer Betracht bleiben.

- Phase Datenvorbereitung

  - Aufgabe „Bereinigen von Daten“

    Beispiel:

    Hier ist z.B. im Einzelfall zu klären, wie mit fehlenden Daten in den Quellsystemen umgegangen werden soll und ob bestimmte fehlende Daten die betroffenen Merkmale oder Datensätze für die Stornountersuchung komplett wertlos machen.

  - Aufgabe „Integrieren von Daten“

    Beispiel:

    Werden die Vertragsdaten in unterschiedlichen Systemen geführt (z.B. wegen noch nicht abgeschlossener Migration), müssen die Daten vor der Entwicklung des Stornomodells zusammengeführt und damit insbesondere hinsichtlich Datenformaten und Bezeichnern vereinheitlicht werden.

- Phase Modellierung

  - Aufgabe „Auswählen der Modellbildungsverfahren“

    Beispiel:

    Bei der Stornoprognose handelt es sich um ein Klassifikationsproblem, für das sich verschiedene Verfahren anbieten (z.B. GLM,

LASSO, künstliches Neuronales Netz). Hier ist eine Auswahl der erfolgversprechendsten Verfahren vorzunehmen, die im Folgenden weiter untersucht werden sollen.

- Aufgabe „Bewerten des Modells“

  Beispiel:

  Hier werden bereits für bestimmte Hyperparameter gerechnete Modelle anhand geeigneter Gütekriterien bewertet und untereinander verglichen.

- Phase Evaluierung
  - Aufgabe „Evaluieren der Ergebnisse“

    Beispiel:

    Bei dieser Aufgabe sind zum einen für das gewählte Modell umfassende Gütekriterien (wie z.B. Genauigkeit, Sensitivität, etc.) zu bestimmen und mittels sinnvoller Schwellenwerte zu interpretieren (sog. Validierungskriterien). Zum anderen ist zu bewerten, inwiefern die in der ersten Phase für die Stornountersuchung definierten Data-Mining-Ziele erreicht wurden.

  - Aufgabe „Bestimmen der nächsten Schritte“

    Beispiel:

    Abhängig von den gefundenen Ergebnissen könnte zum Beispiel eine stärkere Fokussierung auf bestimmte (stornoarme) Produktgruppen vorgenommen werden oder eine Optimierung des Beschwerdemanagements vorgesehen werden.

- Phase Bereitstellung / Anwendung
  - Aufgabe „Planen der Bereitstellung“

    Beispiel:

    Hier ist festzulegen, wie die „nächsten Schritte“ konkret umzusetzen sind und welche Zielgruppen (z.B. Produktentwicklung, Bestandsführung, Vertrieb) in welcher Form von den Ergebnissen der Stornountersuchung zu informieren sind.

  - Aufgabe „Planen von Überwachung und Anpassung“

Beispiel:

Hier ist festzulegen, wie die vorgeschlagenen Maßnahmen zur Stornoverhütung nachverfolgt werden können und wie eine kontinuierliche Überprüfung der Ergebnisse in den Folgejahren erfolgen soll.

Aufgabe 7. *[4.1 Überwachtes Lernen; 3.3 Visualisierung] [20 Punkte]*

(a) *[5 Punkte]* Zur Schätzung einer Schadenhöhe s soll das (einfache) lineare Modell

$$s = 200 + 1.5 \cdot x + 10 \cdot y + 1200 \cdot z$$

verwendet werden. Damit ist die Schadenhöhe s für die folgenden vier Personen zu ergänzen.

Hinweis: Die Tabelle und die berechneten Ergebnisse sind auf die ausgeteilten Arbeitsblätter zu übertragen!

id	x	y	z	s
1	1	2	0	
2	0	2	5	
3	0	6	1	
4	1	1	2	

(b) [5 Punkte] Wie sieht ein Lorenz-Plot in den Extremausprägungen aus? Zeichnen und kommentieren Sie unter Verwendung der in Teil (a) angegebenen Daten die beiden Extremausprägungen der Lorenzkurve.

(c) [10 Punkte] Zeichnen Sie die Lorenz-Kurve für die Daten aus Teil (a). Berechnen Sie den Gini-Koeffizienten. Wie ist dieser zu interpretieren?

Hinweis: Bei den Rechnungen ist auf drei Nachkommastellen zu runden.

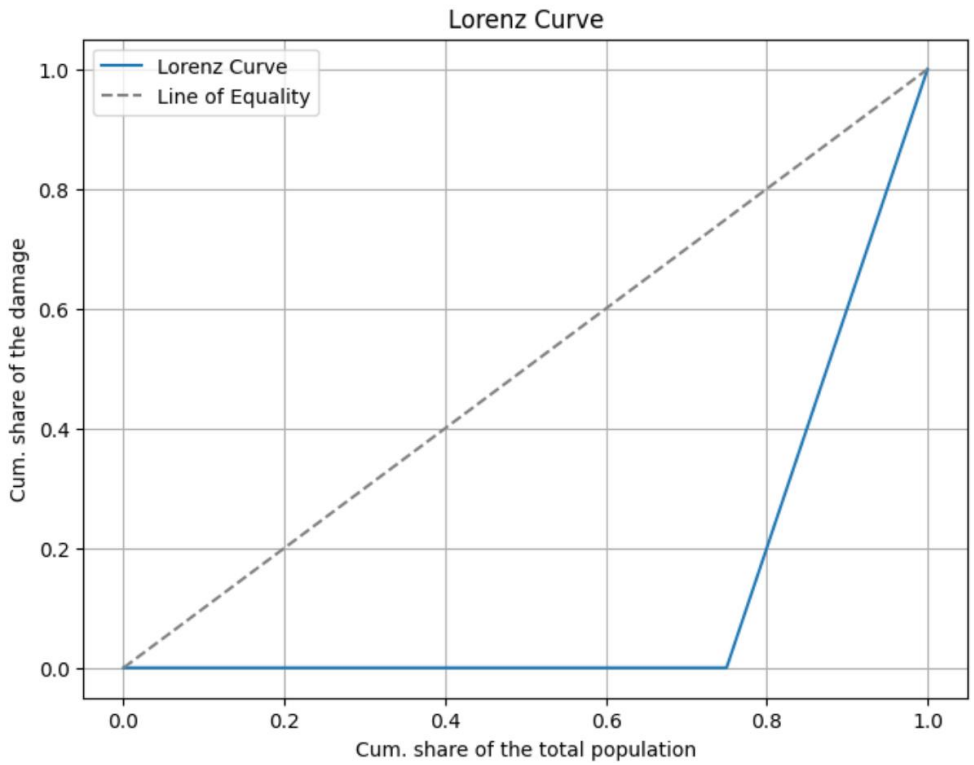
Lösungsvorschlag:

(a) Die berechneten Werte ergeben sich wie folgt:

id	x	y	z	s

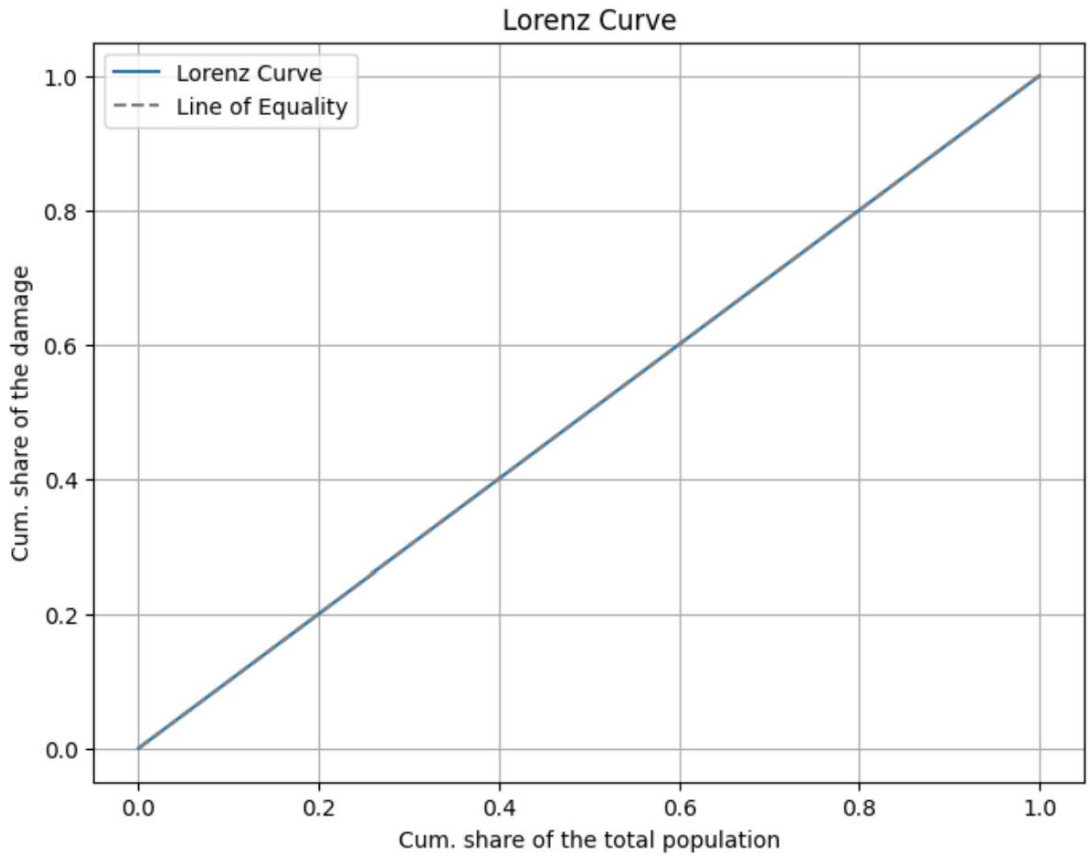
1	1	2	0	221,5
2	0	2	5	6.220,0
3	0	6	1	1.460,0
4	1	1	2	2.611,5

(b) Im ersten Extremfall liegt nur eine Person mit einem Gesamtschaden von 10.513 vor. In diesem Fall ergibt sich folgendes Bild der Lorenz-Kurve:



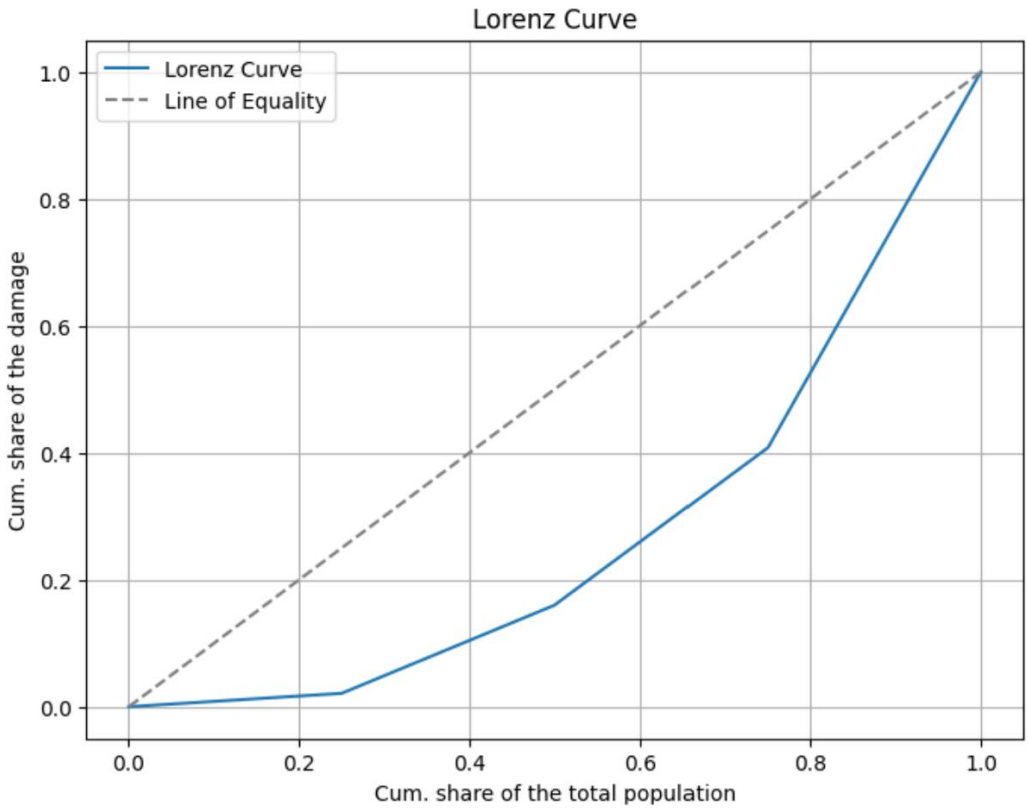
Daraus ergibt sich ein Gini-Koeffizient von 1 - 2·0,5 · 0,25 = 0,75. Da wir nur endlich viele Beobachtungen haben, ist $G < 1$.

Im zweiten Extremfall wird die Gesamtschadenhöhe auf alle vier Personen gleichverteilt (2.628,25). Damit entspricht die Lorenzkurve exakt der Winkelhalbierenden:



Damit ergibt sich ein Gini-Koeffizient von 0.

(c) Für die Daten aus Teil (a) ergibt sich folgende Lorenz-Kurve:



Dabei wurden die folgenden Punkte für die Visualisierung verwendet (gerundet auf drei Nachkommastellen):

x-Anteil	0	0,25	0,5	0,75	1
y-Anteil	0	0,021	0,16	0,408	1

Daraus ergibt sich der Flächeninhalt unter der Lorenz-Kurve als ca. 0.272, und damit der Gini-Index zu 0,456.

Verglichen mit den möglichen Extremwerten ergibt sich eine mittlere Ungleichverteilung.

Aufgabe 8. *[4.2 Deep Learning 2] [30 Punkte]*

Für die Identifikation von Objekten in Bildern soll ein Convolutional Neural Network (CNN) genutzt werden. Dieses wird auf einem Datensatz trainiert, der aus insgesamt 60.000 farbigen Bildern besteht, wobei jedes Bild in der Auflösung 32x32 vorliegt. Insgesamt sind im Datensatz 10 Klassen vorhanden, wobei von jeder Klasse insgesamt 6.000 Bilder vorliegen.

(a) [5 Punkte] Zunächst sind die folgenden Elemente eines CNNs zu beschreiben. Gehen Sie dabei, falls eine Dimensionsreduktion erfolgt, auch auf die zugehörigen Formeln ein.

(i) Convolutional Layer

(ii) Aktivierungsfunktion

(iii) Pooling Layer

(iv) Flatten Layer

(v) Dense Layer

(b) [12 Punkte] Für die Verarbeitung der in der Einleitung angesprochenen Bilder soll folgendes Netz verwendet werden:

```
model = models.Sequential()

model.add(layers.Conv2D(filters=32, kernel_size=(3, 3), strides=(1,1),
use_bias=True, activation='relu', input_shape=(32, 32, 3)))

model.add(layers.MaxPooling2D(pool_size=(2, 2), strides=(2, 2), pad-
ding='valid'))

model.add(layers.Conv2D(filters=64, kernel_size=(3, 3), strides=(1,1),
use_bias=True, activation='relu'))

model.add(layers.MaxPooling2D(pool_size=(2, 2), strides=(2,2), pad-
ding='valid'))

model.add(layers.Conv2D(filters=64, kernel_size=(3, 3), strides=(1,1),
use_bias=True, activation='relu'))

model.add(layers.Flatten())

model.add(layers.Dense(units=64, activation='relu'))

model.add(layers.Dense(units=10, activation=None))
```

Zur Bearbeitung der folgenden Aufgabe gehen sie von einer Batch-Size von 1 aus, d.h. es wird nur ein einzelnes Bild betrachtet. Berechnen und erläutern Sie, ausgehend von einem Bild der Auflösung 32x32x3, für jede Schicht einzeln die

- Inputdimension
- Anzahl der in der Schicht verwendeten trainierbaren Parameter
- Ausgabedimension

Wie viele trainierbare Parameter werden in diesem Modell insgesamt verwendet?

(c) [10 Punkte] Basierend auf der letzten Schicht im Modell aus Teil (b): Was ist bei der Loss-Funktion zu berücksichtigen bzw. welche sollte verwendet werden und wie wird diese berechnet? Gehen Sie dabei davon aus, dass die Labels in der Form [0 = 'Objekt0', ..., 9='Objekt9'] vorliegen.

Erläutern Sie zwei gängige Strategien zur Wahl der Anzahl der Filter pro Layer in einem CNN.

(d) [3 Punkte] Nach dem Training des Modells unter Verwendung der eingangs genannten Trainingsdaten, dem Modell aus Teil (b) sowie der Loss-Funktion aus Teil (c) wird das Training gestartet. Im Anschluss daran erhalten Sie die folgende Graphik:

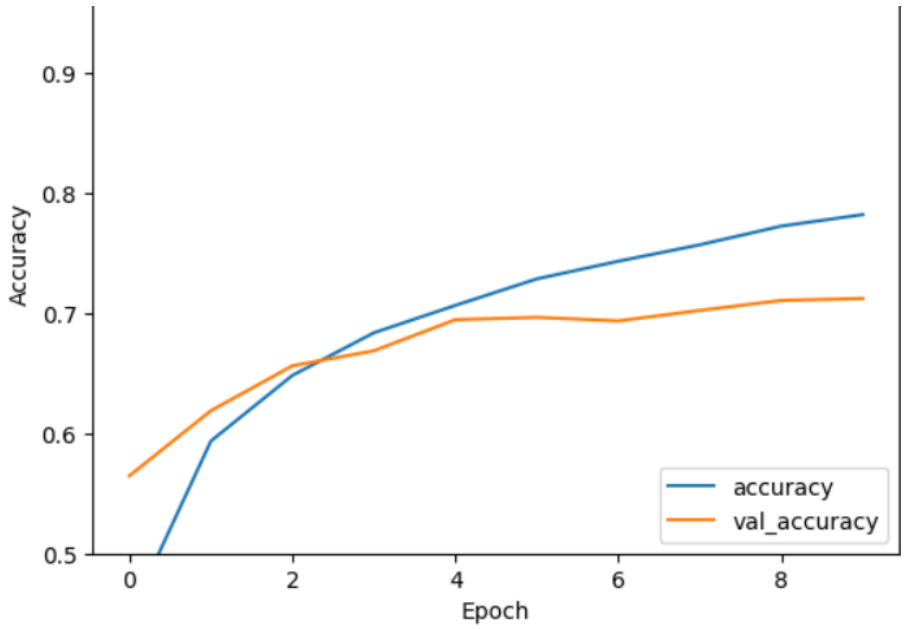


Abbildung 1: (Training) accuracy und validation accuracy

Was können Sie anhand dieser Graphik zu dem Modell aus Teil (b) sagen? Wie ist dieses im Vergleich zu den Bildern (a) und (b) (siehe Abbildung 2) einzuordnen?

Abbildung 2: Vergleichsgraphik (a) links und (b) rechts

Lösungsvorschlag:

(a) Convolutional Layer: Dieser führt eine Faltung durch, d.h. eine gewichtete Summe über lokale Bereiche der Eingabe. Das passiert mit Hilfe eines (kleinen) Filters (Kernels). Dieser gleitet über das Eingabebild wie ein Fenster, und erkennt dabei bestimmte Muster. Je mehr Filtertypen verwendet werden, desto mehr Merkmalstypen kann das Modell lernen. Dabei gibt die Schrittweite (Stride) an, wie viele Pixel der Filter pro Schritt springt. Das Padding gibt an, ob und wie die Ränder des Bildes behandelt werden: Dabei bedeutet 'valid', dass kein Padding verwendet wird (der Output wird kleiner), 'same' bedeutet, dass ein Zero-Padding durchgeführt wird (d.h. der Output behalt die gleiche Größe wie der Input, falls ein Stride von 1 gewählt wurde).

Bei einem Bild mit den Dimensionen $H\ x\ W$, Kernel (k_H, k_W), Padding (p_H, p_W) und Stride (s_H, s_W) ergeben sich die neuen Dimensionen zu:

$$H_{out} = \left\lfloor \frac{H\ -\ k_H + 2\ p_H}{s_H} + \ 1 \right\rfloor$$

sowie

$$W_{out} = \left\lfloor \frac{W\ -\ k_W + 2\ p_W}{s_W} + \ 1 \right\rfloor.$$

Aktivierungsfunktion: Analog der Funktionsweise in Dense-Layers sind diese u.a. für die Nichtlinearitäten verantwortlich und folgen typischerweise direkt auf einen convolutional Layer.

Pooling Layer: Hierdurch werden die räumlichen Dimensionen der Merkmalsarten reduziert. Dies bedeutet weniger Parameter und Berechnungen im Netz, was dieses schneller macht und eine Überanpassung verhindert. Die Dimension des Ausgangsbilds hängt davon ab, welche Paddingmethode verwendet wird. Exemplarisch für padding='valid' ergeben sich (unter Berücksichtigung eines Pooling-Fensters der Größe $(pool_H, pool_W)$):

$$H_{out} = \left\lfloor \frac{H_{in} - pool_H}{s_H} \right\rfloor + 1$$

sowie

$$W_{out} = \left\lfloor \frac{W_{in} - pool_W}{s_W} \right\rfloor + 1$$

Flatten-Layer: Wandeln die 2D-Feature Maps in einen eindimensionalen Vektor um. Dieser Schritt dient als Vorbereitung für die Verarbeitung in Dense-Layern.

Dense-Layer: Verarbeiten die extrahierten Merkmale und führen eine Klassifikation oder Regression durch.

(b) Für den ersten Conv2D-Layer besteht die Eingabe aus einem Bild der Dimension 32x32x3. Insgesamt werden 32 Filter verwendet, und der Kernel hat eine Größe von 3x3. Aus der angegebenen Formel aus Teil (a) ergibt sich damit ein Ausgabeformat von 30x30x32. Pro Kernel werden 3x3x3=27 Parameter verwendet, daraus ergeben sich 27x32=864 Parameter für die 32 Kernels. Insgesamt (inkl. Bias) damit 864+32=896.

Der zweite Layer (MaxPooling) enthält keine trainierbaren Parameter. Nach der Formel unter (a) ergibt sich ein Ausgabeformat von 15x15x32.

Für den dritten Layer (Conv2D) ergeben sich mit analogen Überlegungen wie für Layer 1 insgesamt 18.496 trainierbare Parameter und ein Ausgabeformat von 13x13x64.

Für den vierten Layer (MaxPooling) sind keine Parameter zu berücksichtigen. Da kein Padding berücksichtigt wird, wird die letzte Zeile/Spalte nicht mit berücksichtigt. Als Ausgabeformat ergibt sich 6x6x64.

Für den fünften Layer (Conv2D) ergeben sich 36.928 Parameter. Als Ausgabeformat ergibt sich 4x4x64.

Der sechste Layer (flatten) wandelt das Ausgabeformat 4x4x64 in einen eindimensionalen Vektor mit insgesamt 1.024 Elementen um.

Der siebte Layer (Dense) enhält insgesamt 1024x64+64 = 65.600 trainierbare Parameter, und übergibt einen eindimensionalen Vektor mit 64 Elementen.

Der finale Layer (Dense) enthält insgesamt 64x10+10 = 650 trainierbare Parameter.

Damit enthält das Netzwerk insgesamt 122.570 trainierbare Parameter.

(c) Insgesamt ist aus 10 vorgegebenen Klassen eine Klasse zu wählen. Es ist zu berücksichtigen, dass der letzte Layer keine Aktivierungsfunktion angegeben hat (activation=None). Zudem sind die Label nicht One-Hot codiert, sondern liegen als ganze Zahlen (z.B. 2, 5, etc.) vor. In diesem Szenario eignet sich SparseCategoricalCrossentropy(from_logits=True) als Loss-Funktion. Diese funktioniert wie folgt:

Zunächst wird ein Softmax auf Basis der „rohen“ Ausgabewerte durchgeführt. Daraus errechnet sich der Loss wie folgt: $Loss = -\log(p_{richtig})$, wobei $p_{richtig}$ die Wahrscheinlichkeit der richtigen Klasse bezeichnet.

Zwei mögliche Strategien:

1) Progressive Erhöhung der Filterzahl pro Layer (z.B. 32 -> 64 -> 128)
   Die Motivation liegt darin, dass in frühen Layern einfache Muster (wie z.B. Kanten oder Texturen) erkannt werden. In späteren Layern werden die Merkmale

komplexer, was mehr Filter notwendig macht, um mehr Merkmalskombinationen abzudecken.

2) Konstante Anzahl von Filtern (z.B. 64 -> 64 -> 64)
Hierdurch kann eine gleichmäßige Rechenlast erzeugt werden. Zudem ist die Architektur entsprechend einfacher. Dieses Vorgehen ist ggf. ausreichend für einfache Aufgaben.

(d) Aus Abbildung 1 ist folgendes Verhalten ersichtlich: Sowohl Trainings- als auch Validation-Accuracy steigen zunächst an. Ab ca. Epoche 3/4 steigt die Trainings-Accuracy weiter an, während die Validation-Accuracy Kurve sinkt bzw. abflacht, was ein Hinweis auf Overfitting ist. Dagegen sieht man in Abbildung 2 (a) ein typisches Underfitting (Sowohl Trainings- als auch Validierungs-Accuracy bleiben niedrig. Das Modell hat nicht genug Kapazität oder trainiert nicht lange genug) und in Teil (b) einen guten Fit bzw. eine gute Generalisierung (Training und Validierung steigen gleichmäßig an und es ist kaum Abstand zwischen beiden Kurven zu sehen, d.h. das Modell verallgemeinert gut).

DAV
Deutsche
Aktuarvereinigung e.V.

Written examination in the subject

Actuarial Data Science Advanced

in accordance with examination regulations 5
of the German Actuarial Association e. V.

on October 24, 2025

Notes:

- A pocket calculator is permitted as an aid.
- The total score is 180 points. The exam is passed if at least 90 points are achieved.
- Please check the exam you have received for completeness. The exam consists of 36 pages.
- All answers must be justified and the solution to math problems must be clear.
- Please avoid the unrelated scattering of solutions to the individual parts of the task when creating the solution.
- For reasons of better readability, the language forms male, female and diverse (m/f/d) are not used simultaneously.

Members of the examination board:

Axel Kiermaier, Dr. René Külheim, Prof. Dr. Jonas Offtermatt

Task 1 *[1.1 Social environment & ethics 2, 3.4 Innovative products] [22 points]*

You work as a data scientist and actuary at **SecureEverything AG**. Like every company today, your company faces the challenge of meeting new regulatory requirements, dealing with disruptive market changes and developing innovative products in order to remain competitive. The Management Board has asked you to prepare an analysis and draw up specific recommendations for the corporate strategy.

Note: Your client is your board of directors. So write management summaries and not scientific papers.

(a) *[6 points]* The Board would like to know how AI systems are classified within the EU's Artificial Intelligence Act and your reasoned(!) assessment of which category automated claims settlement using AI falls into.

(b) *[4 points]* The Management Board is concerned about potential disruption risks arising from technological developments and new market participants. Name and explain two specific risks of disruption that could arise for traditional insurers such as SecureEverything AG.

(c) *[6 points]* In order to keep up with the competition, new innovative products are to be developed using data science. Name three advantages that data science products can have for insurers or customers.

(d) *[6 points]* After you have explained the three areas of action (increased regulation, risk of disruption, new innovative products on the market) to the Management Board, they ask you to propose recommendations for action for SecureEverything AG. Briefly and concisely name three specific recommendations based on the fields of action that you think the company should follow in the near future.

Proposed solution:

(a) The EU's Artificial Intelligence Act (AI Act) classifies AI systems into four risk levels:

- Inadmissible risk (e.g. social scoring)
- High risk (e.g. lending)
- Limited and minimal risk (e.g. chatbots, recommendation systems)
- No Risk

Automated claims settlement using AI is likely to fall into the "high risk" category, as it falls into the sensitive area of financial services and can have a significant impact on consumers. Strict compliance with transparency and fairness requirements is necessary.

(1 point per risk class, 2 for the justification of the claim settlement)

(b) Two disruption risks would be, for example:

- InsurTech start-ups & BigTech entry: Technology-driven start-ups and large technology companies (e.g. Google, Amazon) could enter the insurance market with data-based business models.
- Pay-as-you-go & on-demand insurance: New flexible models based on real-time data (telematics, IoT) are threatening traditional policies. Customers want situational, personalized policies instead of long-term contracts.

(2 points per risk of disruption, other entries are possible)

(c) Advantages and possibilities of using data science:

- Development of detailed data bases for risk assessment
- More precise differentiation and selection of risks
- Improved ability to assess and analyze underwriting risks (including fraud detection and loss prevention)
- Development of detailed data bases for evaluating the customer relationship
- Improved analysis and modeling of the customer relationship
- Optimization of customer group management and database marketing
- ...

(2 points per named advantage up to a maximum of 6 points, other entries are possible)

(d) Creativity and management thinking are required here. There is therefore no concrete right or wrong. Possible answers would be:

- Strategically integrate regulatory compliance: Early adaptation to the AI Act and GDPR guidelines to avoid risks. Installation of an AI Officer?
- Focus on data-driven innovations: Development of personalized tariffs & AI-supported risk models for differentiation in the market. Deployment of an actuarial data science team?
- Cooperation with InsurTechs & technology companies: Partnerships with startups or tech companies to strengthen innovation and digital capabilities. Participation in innovation networks?

(2 points per recommended action)

Task 2 *[3.3 Visualization 1] [20 points]*

SecureEverything AG analyzes motor insurance claims in order to optimize rates. You receive a random sample of **10 claims** with the characteristics of **driver age, vehicle type, claim amount** and **mileage**.

Driver age	Vehicle type	Amount of damage	Mileage
22	Small car	0	50.000
45	SUV	5.000	NaN
33	Combi	0	80.000
60	SUV	7.500	120.000
29	Small car	0	30.000
50	Small car	1.300	20.000
38	Combi	3.200	75.000
27	SUV	6.000	100.000
41	Small car	1.800	45.000
55	Combi	2.200	NaN

Create a suitable graphical representation for each of the following three questions (interpretation of the graphs is not required):

(a) [8 points] Is there a correlation between vehicle type and the probability of damage?

(b) [7 points] What influence does the vehicle type have on the amount of damage?

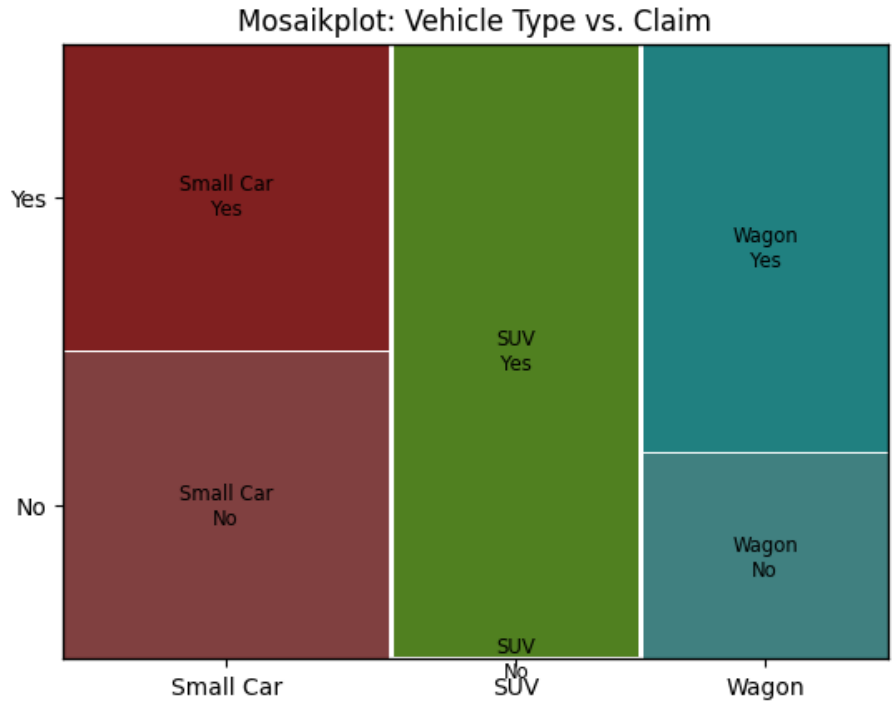
(c) [5 points] What is the distribution of cases with damage and without damage?

Hint:
Create **at least one mosaic plot and one simplified box plot (without quantile values)** to answer the questions. In addition, choose other suitable chart types to answer the questions. No justification is required, only suitable graphs.

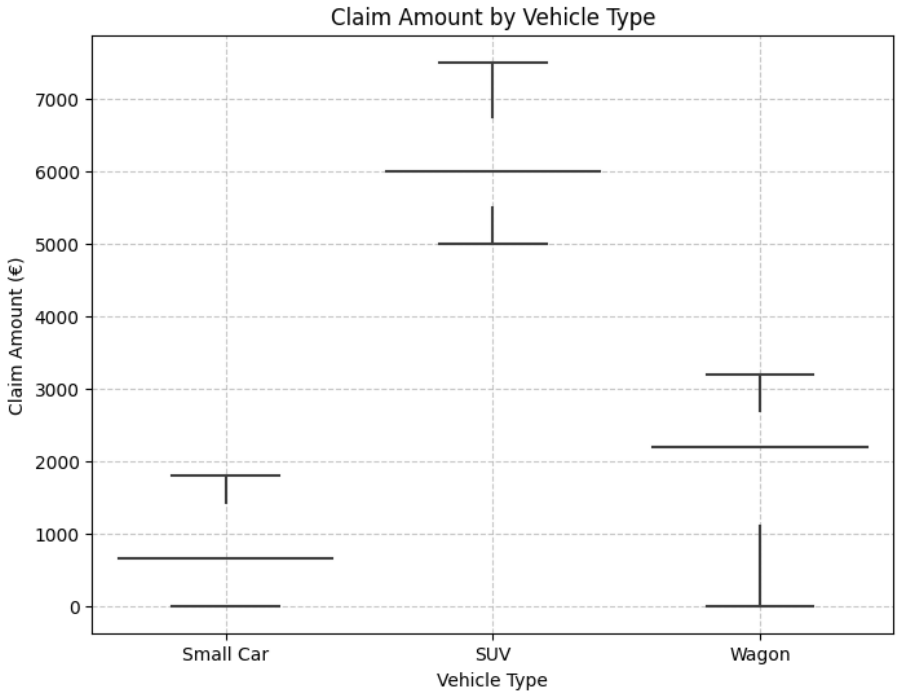
Please remember the basic requirements for visualizations: Use ruler, label axes, legends, titles, etc.

Proposed solution:

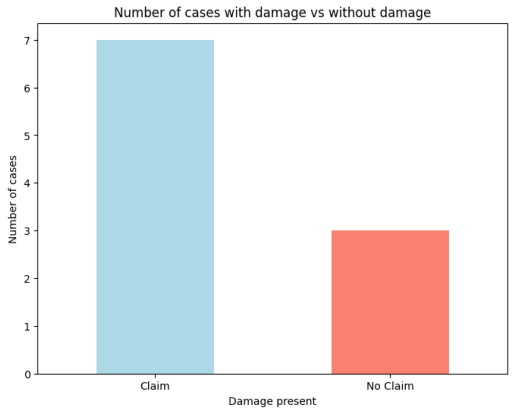
(a) Possible solution a mosaicplot:

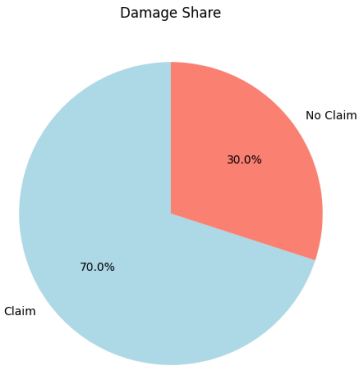


(b) Possible solution a simplified boxplot:



(c) Possible solution a bar or pie chart:





Task 3 *[4.4 Data preparation for model creation, 5.1 Programming languages for data science] [20 points]*

In this task, you will again work with the data set of claims from a car insurance company from task 2. Assume that the data has been transferred to a Pandas dataframe using the following code:

```
# Create DataFrame

data = {

    'Driver_Age': [22, 45, 33, 60, 29, 50, 38, 27, 41, 55],

     'Vehicle_Type': ['Small Car', 'SUV', 'Wagon', 'SUV', 'Small Car',
'Small Car', 'Wagon', 'SUV', 'Small Car', 'Wagon'],

     'Claim_Amount': [0, 5000, 0, 7500, 0, 1300, 3200, 6000, 1800,
2200],

      'Mileage': [50000, None, 80000, 120000, 30000, 20000, 75000,
100000, 45000, None]

}



dfe = pd.DataFrame(data)
```

Please enter the output of the print command for each of the Python code lines below

(a) *[5 points]*

```
suv_dfe=dfe[dfe['Vehicle_Type']=='SUV'].reset_index(drop=True)

print(suv_dfe)
```

(b) *[5 points]*

```
age_over_30 = dfe[dfe['Driver_Age'] < 30]

print(age_over_30[['Driver_Age', 'Vehicle_Type']])
```

(c) *[5 points]*

```
dfe['DamageQuota'] = dfe['Claim_Amount'] / dfe['Mileage']

print(dfe.iloc[7])
```

(d) *[5 points]*

```
dfe['Mileage']                                                    =
dfe['Mileage'].fillna(dfe['Mileage'].median())

print(dfe['Mileage'])
```

Proposed solution:

(a) Please note that the index must be entered correctly here. This will be regenerated.

```
   Driver_Age Vehicle_Type  Claim_Amount   Mileage
0          45          SUV          5000       NaN
1          60          SUV          7500  120000.0
2          27          SUV          6000  100000.0
```

(b) Attention, variable name does not match the "<" operator.

```
   Driver_Age Vehicle_Type
0          22    Small Car
4          29    Small Car
7          27          SUV
```

(c) Only the eighth row of the table (Python starts at 0).

```
Driver_Age              27
Vehicle_Type           SUV
Claim_Amount          6000
Mileage           100000.0
DamageQuota           0.06
Name: 7, dtype: object (This line does not have to be specified for full points)
```

(d) The entire column must be output, not just the imputed values. The last row and the index do not have to be specified for the full score.

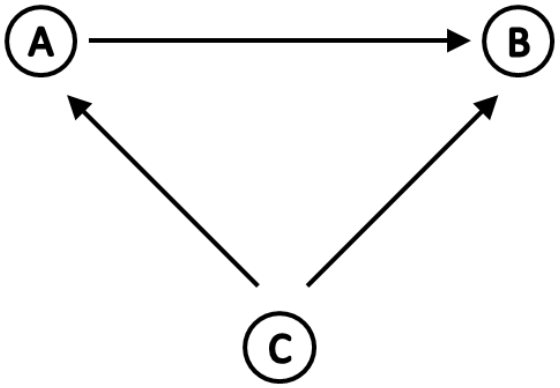
```
0     50000.0
1     62500.0
2     80000.0
3    120000.0
4     30000.0
5     20000.0
6     75000.0
7    100000.0
8     45000.0
9     62500.0
Name: Mileage, dtype: float64
```
(This line does not have to be specified for the full score)

Task 4 *[4.3 Correlation & causal inference] [19 points]*

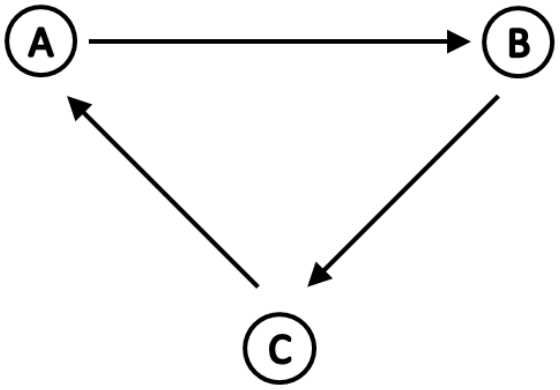
(a) *[12 points]* Explain the concepts of survival bias, outcome bias and omitted variable bias and give an example of each from the insurance industry. Also answer what each of these concepts has to do with the topic of causality.

(b) *[7 points]* In the following four graphs (b1) - (b4), the causal relationship between *A* and *B* takes center stage. What role does *C* play in this context? Determine the joint probability distribution *P(A,B,C)* in each case.

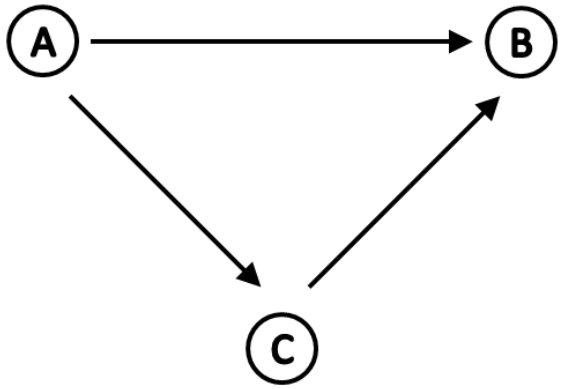
(b1):



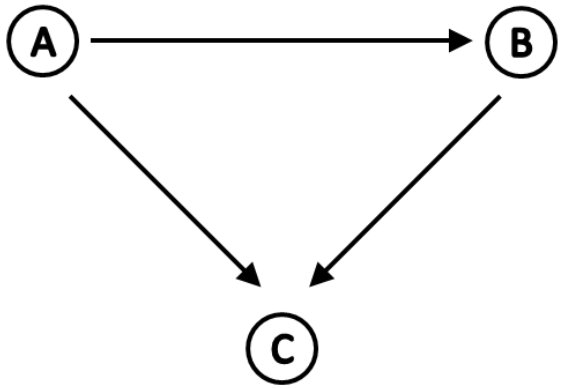
(b2):



(b3):



(b4):



Proposed solution:

(a) Survival bias, outcome bias and omitted variable bias are different forms of statistical bias.

1. Survival bias

Survival bias or survivor bias consists of systematically overestimating the probability of survival or success because survivors/successful people are more likely to appear in a statistic than unsuccessful people.

Example: Analysis of the satisfaction of participants in a bonus program that has been running for a long time. In this case, most of the dissatisfied participants have already canceled in the past, which means that the proportion of satisfied participants among the "living" participants is too high.

Link to causality: Survival bias only considers selective and therefore biased samples. This means that the correlations and true causes cannot be identified in certain analyses.

2. Outcome bias

Outcome bias occurs when the quality of a decision is judged on the basis of its outcome. If the result is positive, the decision is subsequently classified as good, possibly according to the principle of "the result justifies the means"; possible risks are thus ignored. In contrast, if the result is negative, the decision is retrospectively classified as bad, even though it was the best possible one based on the information available at the time of the decision.

Example: The underwriting of an application led to its acceptance; later, a high loss occurs under the contract. A subsequent assessment by the underwriter based on the result is not necessarily adequate.

Connection to causality: In the case of outcome bias, it is assumed that there is a causal connection between the decision and the outcome; any uncertainties that may exist at the time of the decision are disregarded. As a result, possible risks of the decision are ignored. This means, for example, that a decision is presented as positive, even though the positive result did not occur as a causal effect, but merely as a result of fortunate circumstances.

3. Omitted Variable Bias

Omitted variable bias occurs when one or more relevant variables are omitted.

Example: A company identifies higher accident frequencies with red vehicles and therefore considers the color as a risk factor, but not the driving behavior (for which

no data is available). In fact, a certain color preference can correlate with a certain driving behavior, but is not the cause of it and therefore not the cause of a higher accident frequency.

Link to causality: Confounders not taken into account in a DAG (such as the lack of information on driving behavior in the example due to a lack of telematics) lead to unfounded causal statements.

(b) In **(b1)**, *C* is a confounder for the effect of *A* on *B*. It influences both *A* and *B*. A confounder can lead to incorrect conclusions if it is not adequately controlled or taken into account.

$$P(A,B,C) = P(B|A,C)\,P(A,C) = P(B|A,C)\,P(A|C)\,P(C)$$

There is no DAG in **(b2).** The graph is cyclical and therefore not suitable for mapping a causal relationship.

In particular, $P(A,B,C)$ cannot be factorized in a similar way.

In **(b3)**, the effect of *A* on *B* is not (only) direct, but partly via *C. (Note: C is also referred to here as a mediator.)* To determine the direct effect of *A* on *B*, the total effect of *A* on *B* (e.g. by regression) and the mediation effect of *C* (e.g. by path analysis) could be determined separately using suitable methods and the difference calculated from this.

$$P(A,B,C) = P(B|A,C)\,P(A,C) = P(B|A,C)\,P(C|A)\,P(A)$$

In **(b4),** *C* is a collider. Given *C*, *A* and *B* (even if they are independent) become conditionally dependent. This can lead to counterintuitive correlations.

$$P(A,B,C) = P(C|A,B)\,P(A,B) = P(C|A,B)\,P(B|A)\,P(A)$$

Task 5 *[2.1 Data management 2 - Relational DBs] [26 points]*

Preliminary remark: Please use Martin notation (crow's feet notation) throughout this task for ER diagrams.

(a) *[6 points]* Define the first to third normal forms and explain the requirements of each definition using a minimal example from the insurance context in which the respective requirement is violated.

(b) *[12 points]* Develop an ER diagram from the following description of an insurance situation from the field of life insurance. Describe the entities (object types), attributes, relationships and cardinalities and then draw the diagram.

The natural person should be at the center of your insurance. They are described by their first name, surname and date of birth. Persons can be married to each other, which is particularly important for one of your main products (joint life insurance).

In this simplified example, the description of an insurance contract is limited to the sum insured, pro rata premium, payment method, start date and expiration date.

Persons can take on different roles within the framework of an insurance contract: insured person, policyholder, premium payer.

Persons have an address (street, house number, zip code, town).

Model the role and the address as a separate object type.

(c) *[3 points*] Name two advantages and two disadvantages of the explicit modeling of the role as an object type in (b). Which advantage would you emphasize in particular?

(d) *[5 points]* "Translate" the example from subtask (b) into table type descriptions.

Proposed solution:

(a) 1. normal form:

A data model is in the first normal form if it has no multiple properties, i.e. all attributes are atomic.

Counterexample:

A Person table contains the attributes First name, Last name and Telephone numbers. In the latter attribute, several telephone numbers are saved as a comma-separated string, for example

Person_ID	First name	Last name	Telephone numbers
1	Peter	Meier	010/1234567, 089/9876543

Here the attribute telephone numbers is not atomic and therefore the first normal form is violated.

2. normal form:

A data model is in the second normal form if it is in the first normal form and each descriptive property of an object type is functionally dependent on the overall key, but not already on a partial key of this object type.

Counterexample:

A contractual relationship table contains the attributes Contract_ID and Person_ID (which together form the primary key of the table) and Start date, for example:

Contract_ID	Person_ID	Start date
1234567	0000001	01.10.2025
1234567	0000002	01.10.2025

Here, the start date attribute is already clearly dependent on the contract_ID, but this is only a partial key and not the overall key of the table. Therefore, the second normal form is violated.

3. normal form:

A data model is in the third normal form if it is in the second normal form and no descriptive property of an object type is functionally dependent on another descriptive property of this object type.

Counterexample:

A contract table contains the attributes sum insured, product key and product name, for example

Contract_ID	Sum insured	Product key	Product name
1234567	500.000	P001	LV_Classic
1234568	100.000	P001	LV_Classic
1234569	20.000	P002	LV_Funds
1234570	30.000	P002	LV_Funds

Here, the value of the product name attribute is functionally directly dependent on the product key attribute. Therefore, the third normal form is violated.

(b) The following entities and attributes result from the description (here with English identifiers - German identifiers are also possible):

1. Entity "person"

   Attributes:

   - firstName
   - lastName
   - dateOfBirth

2. Entity "address"

   Attributes:

   - street

- houseNumber
- zipCode
- city

3. Entity "contract"

   Attributes:

   - insuranceAmount
   - periodicContribution
   - paymentMethod
   - startDate
   - expiryDate

4. Entity "role"

   Attributes:

   - roleDescription

Description of relationships and cardinalities:

1. address to person

   A person has exactly one address.

   An address can apply to one or more persons (e.g. several persons in one household).

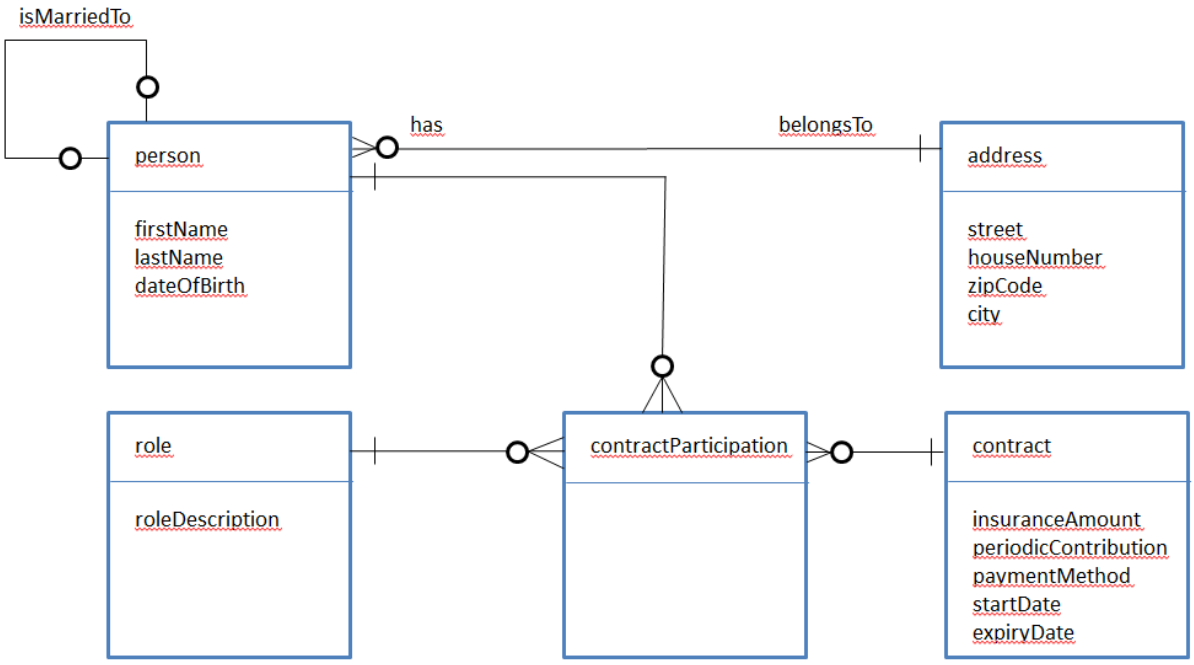
2. person to person (recursive relationship)

   Each person is married to either no one or exactly one other person.

3. contractParticipation (associative entity)

   To reflect the fact that a person can have several different roles in an insurance contract, an associative entity is introduced for the contract participation.

This results in the following ER diagram:



(c) Advantages:

- Normalization
- Flexibility / expandability

Disadvantages:

- Performance effects
- Complexity / oversizing

A particularly important advantage of explicitly modeling the role is the flexibility for future extensions. If, for example, it can be assumed that further roles will be added in the future, this form of modeling offers an invaluable advantage.

(d) The example from subtask (b) can be mapped in table type descriptions as follows:

person (**personID**, firstName, lastName, dateOfBirth, ⇑addressID⇑, *⇑isMarriedTo-personID⇑)*

address (**addressID**, street, houseNumber, zipCode, city)

contract (**contractID**, insuranceAmount, periodicContribution, paymentMethod, startDate, expiryDate)

role (**roleID**, roleDescription)

contractParticipation (⇑ **personID**⇑ + ⇑ **contractID**⇑ + ⇑ **roleID** ⇑)

Task 6 *[3.1 Data Mining 2] [23 points]*

(a) [5 points] Explain in one sentence what CRISP-DM is. What does the abbreviation stand for? Name the individual phases of CRISP-DM.

(b) [18 points] For each phase, name two tasks in the terminology of CRISP-DM and explain these tasks using an example from the insurance environment. Assume a project for lapse prevention throughout.

Proposed solution:

(a) CRISP-DM = Cross Industry Standard Process for Data Mining.

CRISP-DM is a process model for data mining projects that was created in 1999/2000 by a consortium of companies from several industries and is still regarded as the standard model today due to its widespread use.

The 6 phases of CRISP-DM are called

- Business understanding
- Data understanding
- Data preparation
- Modeling
- Evaluation
- Deployment / Application

(b) *Preliminary remark: The following solution is only an example. Other answers are possible. The decisive factor for achieving full marks is the correct naming and assignment of two tasks to each of the individual phases according to CRISP-DM and the description of an example that is as lapse specific as possible.*

- Business understanding phase
    - Task "Determining business objectives"

      Example:

      The company wants to secure the size of its portfolio by reducing the cancellation rate.

    - Task "Determining data mining targets"

      Example

      In order to initiate targeted measures to improve customer loyalty, corresponding targets are set for data mining. For example, a correct prediction of a certain minimum percentage of customers at risk of canceling should be achieved.
- Data understanding phase

- Task "Collecting original data"

  Example:

  The contract database(s) and customer interactions (correspondence, calls) are particularly suitable data sources here, as is information on the market environment (such as the interest rate situation).

- Task "Checking the data quality"

  Example:

  The completeness of the data is important, for example. If, for historical reasons, the selected data sources do not contain all contracts, for example, contracts with cancellation behavior that differs from the rest could be excluded.

- Data preparation phase
  - Task "Clean up data"

    Example:

    In individual cases, for example, it must be clarified how to deal with missing data in the source systems and whether certain missing data renders the affected characteristics or data records completely worthless for the reversal investigation.

  - Task "Integrating data"

    Example:

    If the contract data is managed in different systems (e.g. because migration has not yet been completed), the data must be merged before the cancellation model is developed and thus standardized, particularly with regard to data formats and identifiers.

- Modeling phase
  - Task "Selecting the modeling process"

    Example:

    Lapse prediction is a classification problem for which various methods are available (e.g. GLM, LASSO, artificial neural network). A

selection of the most promising methods should be made here, which will be examined further below.

- Task "Evaluating the model"

  Example:

  Models that have already been calculated for certain hyperparameters are evaluated and compared with each other using suitable quality criteria.

- Evaluation phase
  - Task "Evaluating the results"

    Example:

    In this task, comprehensive quality criteria (such as accuracy, sensitivity, etc.) must be determined for the selected model and interpreted using meaningful threshold values (so-called validation criteria). Secondly, the extent to which the data mining objectives defined in the first phase for the reversal analysis have been achieved must be evaluated.

  - Task "Determine the next steps"

    Example:

    Depending on the results found, a stronger focus could be placed on certain (low-cancellation) product groups, for example, or an optimization of complaint management could be planned.

- Deployment / application phase
  - Task "Planning the provision"

    Example:

    Here, it is necessary to determine how the "next steps" are to be implemented in concrete terms and which target groups (e.g. product development, inventory management, sales) are to be informed of the results of the cancellation investigation and in what form.

  - Task "Planning monitoring and adaptation"

    Example:

It must be determined here how the proposed measures to prevent lapses can be tracked and how the results are to be continuously reviewed in subsequent years.

Task 7 *[4.1 Supervised learning; 3.3 Visualization] [20 points]*

(a) *[5 points]* To estimate the claims amount s, the (simple) linear model

$$s = 200 + 1.5 \cdot x + 10 \cdot y + 1200 \cdot z$$

can be used. The claims amount s must therefore be supplemented for the following four persons.

Note: The table and the calculated results must be transferred to the worksheets provided!

id	x	y	z	s
1	1	2	0	
2	0	2	5	
3	0	6	1	
4	1	1	2	

(b) [5 points] What does a Lorenz plot look like at the extremes? Using the data given in part (a), draw and comment on the two extreme cases of the Lorenz curve.

(c) [10 points] Draw the Lorenz curve for the data from part (a). Calculate the Gini coefficient. How should this be interpreted?

Note: Calculations must be rounded to three decimal places.

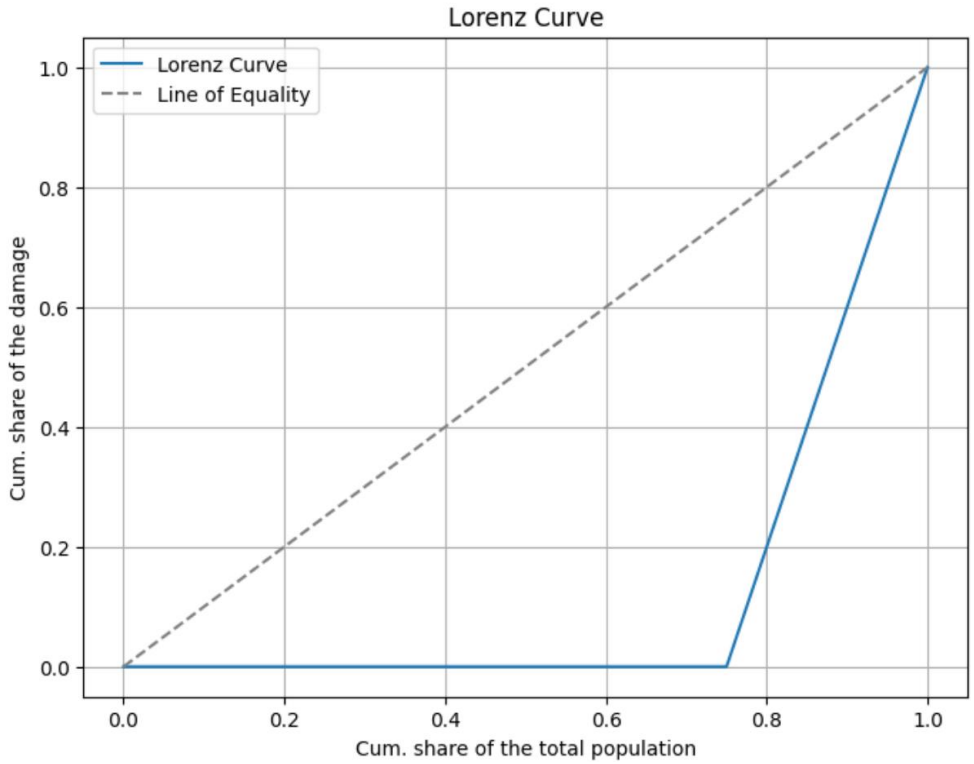
Proposed solution:

(a) The calculated values are as follows:

id	x	y	z	s
1	1	2	0	221,5

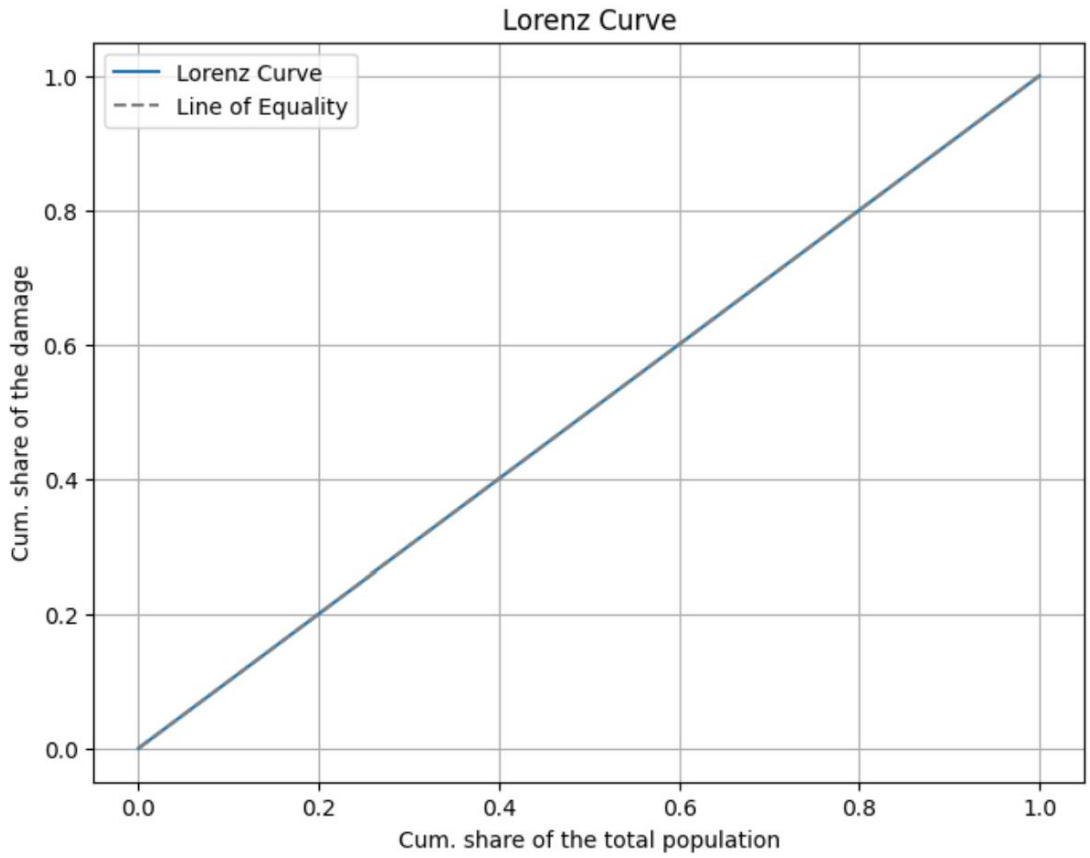
2	0	2	5	6.220,0
3	0	6	1	1.460,0
4	1	1	2	2.611,5

(b) In the first extreme case, there is only one person with a total loss of 10,513. In this case, the Lorenz curve is as follows:



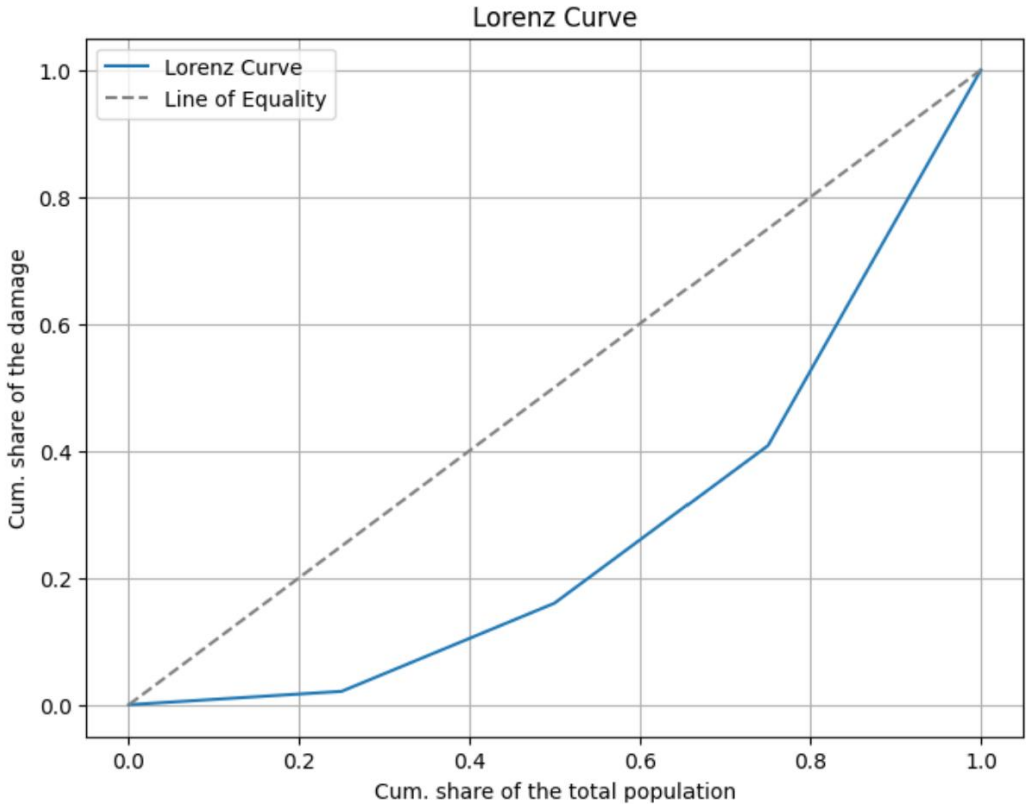
This results in a Gini coefficient of 1 - 2-0.5 - 0.25 = 0.75. Since we only have a finite number of observations,G < 1.

In the second extreme case, the total amount of damage is distributed equally among all four persons (2,628.25). This means that the Lorenz curve corresponds exactly to the bisector:



This results in a Gini coefficient of 0.

(c) The following Lorenz curve results for the data from part (a):



The following points were used for the visualization (rounded to three decimal places):

x-share	0	0,25	0,5	0,75	1
y-share	0	0,021	0,16	0,408	1

This gives the area under the Lorenz curve as approx. 0.272, and thus the Gini index as 0.456.

Compared with the possible extreme values, this results in a medium uneven distribution.

Task 8 *[4.2 Deep Learning 2] [30 points]*

A Convolutional Neural Network (CNN) is to be used to identify objects in images. This is trained on a data set consisting of a total of 60,000 color images, with each image having a resolution of 32x32. There are a total of 10 classes in the data set, with a total of 6,000 images of each class.

(a) [5 points] First, describe the following elements of a CNN. If a dimensional reduction is used, also describe the associated formulas.

(i) Convolutional layer

(ii) Activation function

(iii) Pooling layer

(iv) Flatten Layer

(v) Dense layer

(b) [12 points] The following network should be used to process the images mentioned in the introduction:

```
model = models.Sequential()

model.add(layers.Conv2D(filters=32, kernel_size=(3, 3), strides=(1,1),
use_bias=True, activation='relu', input_shape=(32, 32, 3)))

model.add(layers.MaxPooling2D(pool_size=(2, 2), strides=(2, 2),
padding='valid'))

model.add(layers.Conv2D(filters=64, kernel_size=(3, 3), strides=(1,1),
use_bias=True, activation='relu'))

model.add(layers.MaxPooling2D(pool_size= (2, 2), strides=(2,2),
padding='valid'))

model.add(layers.Conv2D(filters=64, kernel_size=(3, 3), strides=(1,1),
use_bias=True, activation='relu'))

model.add(layers.Flatten())

model.add(layers.Dense(units=64, activation='relu'))

model.add(layers.Dense(units=10, activation=None))
```

To complete the following task, assume a batch size of 1, i.e. only a single image is considered. Based on an image with a resolution of 32x32x3, calculate and explain for each layer individually, the

- Input dimension
- Number of trainable parameters used in the shift
- Output dimension

How many trainable parameters are used in this model in total?

(c) [10 points] Based on the last layer in the model from part (b): What should be considered in the loss function or which one should be used and how is it calculated? Assume that the labels are in the form [0 = 'Object0', ..., 9='Object9'].

Explain two common strategies for selecting the number of filters per layer in a CNN.

(d) [3 points] After training the model using the training data mentioned at the beginning, the model from part (b) and the loss function from part (c), the training is started. You will then receive the following graph:

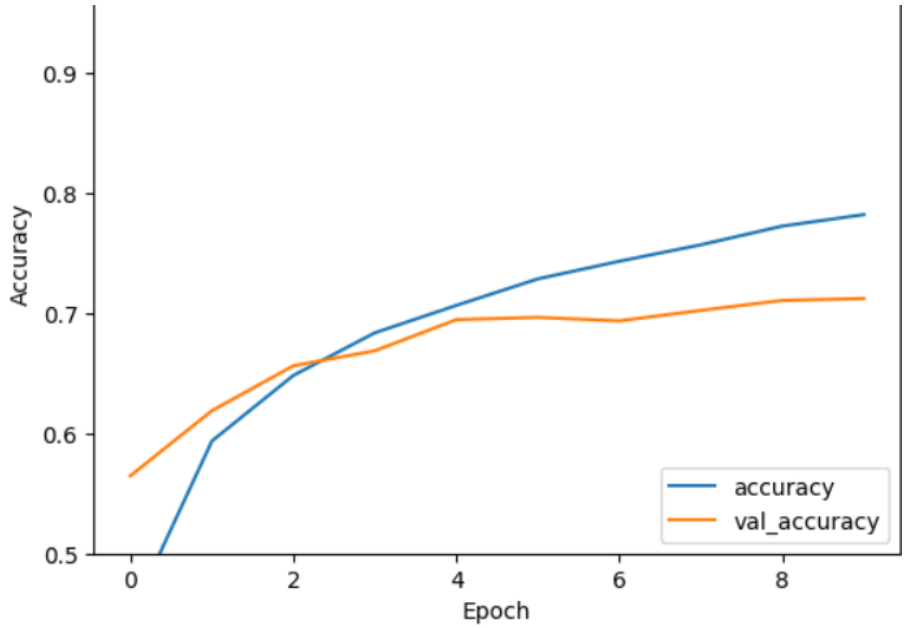


Figure1 : (Training) accuracy and validation accuracy

What can you say about the model in part (b) based on this diagram? How does it compare to images (a) and (b) (see Figure 2)?

Figure 2 Comparison graph (a) left and (b) right

Proposed solution:

(a) Convolutional layer: This performs a convolution, i.e. a weighted sum over local areas of the input. This is done with the help of a (small) filter (kernel). This glides over the input image like a window and recognizes certain patterns. The more filter types are used, the more feature types the model can learn. The step width (stride) indicates how many pixels the filter jumps per step. The padding indicates whether and how the edges of the image are treated: Here, 'valid' means that no padding is used (the output becomes smaller), 'same' means that zero padding is performed (i.e. the output retains the same size as the input if a stride of 1 was selected).

A screen with the dimensions$H\ x\ W$, kernel(k_H, k_W) , padding(p_H, p_W) and stride(s_H, s_W) results the new dimensions:

$$H_{out} = \left\lfloor \frac{H\ -\ k_H + 2\ p_H}{s_H} + \ 1 \right\rfloor$$

and

$$W_{out} = \left\lfloor \frac{W\ -\ k_W + 2\ p_W}{s_W} + \ 1 \right\rfloor.$$

Activation function: Analogous to the mode of operation in dense layers, these are responsible for the non-linearities, among other things, and typically follow a convolutional layer directly.

Pooling layer: This reduces the spatial dimensions of the feature types. This means fewer parameters and calculations in the mesh, which makes it faster and prevents overfitting. The dimension of the output image depends on which padding method is used. Examples for padding='valid' are (taking into account a pooling window of the size):$(pool_H, pool_W)$

$$H_{out} = \left\lfloor \frac{H_{in} - pool_H}{s_H} \right\rfloor + 1$$

and

$$W_{out} = \left\lfloor \frac{W_{in} - pool_W}{s_W} \right\rfloor + 1$$

Flatten layer: Converts the 2D feature maps into a one-dimensional vector. This step serves as preparation for processing in dense layers.

Dense layer: Process the extracted features and perform a classification or regression.

(b) For the first Conv2D layer, the input consists of an image of the dimension 32x32x3. A total of 32 filters are used and the kernel has a size of 3x3. This results in an output format of 30x30x32 from the formula given in part (a). Each kernel uses 3x3x3=27 parameters, resulting in 27x32=864 parameters for the 32 kernels. The total (including bias) is therefore 864+32=896.

The second layer (MaxPooling) does not contain any trainable parameters. The formula under (a) results in an output format of 15x15x32.

For the third layer (Conv2D), the same considerations as for layer 1 result in a total of 18,496 trainable parameters and an output format of 13x13x64.

No parameters need to be taken into account for the fourth layer (MaxPooling). As no padding is taken into account, the last row/column is not included. The output format is 6x6x64.

There are 36,928 parameters for the fifth layer (Conv2D). The output format is 4x4x64.

The sixth layer (flatten) converts the output format 4x4x64 into a one-dimensional vector with a total of 1,024 elements.

The seventh layer (Dense) contains a total of 1024x64+64 = 65,600 trainable parameters, and transfers a one-dimensional vector with 64 elements.

The final layer (Dense) contains a total of 64x10+10 = 650 trainable parameters.

This means that the network contains a total of 122,570 trainable parameters.

(c) A class must be selected from a total of 10 predefined classes. It should be noted that the last layer has no activation function specified (activation=None). In addition, the labels are not one-hot coded, but are available as integers (e.g. 2, 5, etc.). In this scenario, SparseCategoricalCrossentropy(from_logits=True) is suitable as a loss function. This works as follows:

First, a softmax is carried out on the basis of the "raw" output values. The loss is calculated from this as follows: $Loss = -\log(p_{correct})$ where $p_{correct}$ is the probability of the correct class.

Two possible strategies:

1) Progressive increase in the number of filters per layer (e.g. 32 -> 64 -> 128) The motivation is that simple patterns (such as edges or textures) are recognized in early layers. In later layers, the features become more complex, which makes more filters necessary in order to cover more feature combinations.

2) Constant number of filters (e.g. 64 -> 64 -> 64)
   This allows an even computing load to be generated. In addition, the architecture is correspondingly simpler. This procedure may be sufficient for simple tasks.

(d) Figure 1 shows the following behavior: Both training and validation accuracy initially increase. From around epoch 3/4, the training accuracy continues to rise, while the validation accuracy curve falls or flattens out, which is an indication of overfitting. In contrast,Figure 2 (a) shows a typical underfitting (both training and validation accuracy remain low. The model does not have enough capacity or does not train long enough) and in part (b) a good fit or a good generalization (training and validation increase evenly and there is hardly any distance between the two curves, i.e. the model generalizes well).