DAV
Deutsche
Aktuarvereinigung e.V.

**Schriftliche Prüfung im Fach**

**Actuarial Data Science Basic**

gemäß Prüfungsordnung 5
der Deutschen Aktuarvereinigung e. V.

am 23. Mai 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 38 Seiten und einer Anlage.
- 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, Dr. Jonas Offtermatt, Dr. Felix Spangenberg

**Aufgabe 1.** *[1.1 Grundlagen, 3.1 Data Mining 1, 4.2 Unüberwachtes Lernen 1] [36 Punkte]*

Ihr Versicherungsunternehmen plant, seine Produktpalette zu erweitern, um den unterschiedlichen politischen und ökologischen Präferenzen seiner Kunden gerecht zu werden. Dabei sollen nachhaltige Versicherungsprodukte für umweltbewusste Kunden sowie konventionelle Produkte für Kunden, die weniger Wert auf Nachhaltigkeit legen, entwickelt werden. Zur Segmentierung und Analyse der Kundengruppen werden verschiedene Datenpunkte herangezogen, darunter demografische Daten, politische Präferenzen, Fahrzeugtyp, Energieverbrauch, Einkaufsgewohnheiten und Teilnahme an Umweltinitiativen.

(a) *[6 Punkte]* Analysieren Sie die Kundendaten in der folgenden Tabelle:

| | Age | Sex | Income | Political | Vehicle | Energy | Shopping Habits | Environmental |
|---|---|---|---|---|---|---|---|---|
| **1** | 26 | F | 29000.0 | Green | Petrol | Medium | Supermarket | No |
| **2** | 67 | M | 102000.0 | Green | Electric | Low | Organic | Yes |
| **3** | 31 | F | 35000.0 | Center | Diesel | High | Discounter | No |
| **4** | 26 | M | 44000.0 | Conservative | Petrol | Low | Organic | Yes |
| **5** | 57 | M | 72000.0 | Conservative | Diesel | Medium | Discounter | No |
| **6** | 67 | M | 97000.0 | Center | Petrol | Low | Supermarket | No |
| **7** | 24 | F | 18000.0 | Center | Diesel | High | Discounter | No |
| **8** | 74 | M | 106000.0 | Center | Diesel | Low | Discounter | No |

Gehen Sie dabei auf die verschiedenen Datentypen und Merkmalsarten ein.

(b) *[6 Punkte]* Welche Risiken sehen Sie bei diesem Produkt? (Nennen Sie 3 Risiken)

(c) *[8 Punkte]* Zeichnen Sie einen Scatterplot, der das Einkommen (*Income*) der Kunden gegen ihr Alter (*Age*) aufträgt, und dabei die politische Präferenz farblich markiert. Wie viele Cluster vermuten Sie in den Daten?

(d) *[4 Punkte]* Diskutieren Sie, wie sich der Beruf des Aktuars durch die Einführung solcher neuen Produkte ändern könnte. Berücksichtigen Sie dabei die Herausforderungen, die sich aus der Analyse und Nutzung sensibler Kundendaten und Vernetzung weiterer Abteilungen ergeben.

(e) *[12 Punkte]* Führen Sie die unten aufgeführten Schritte eines k-means Clusterings auf den **untenstehenden standardisierten Daten** durch, um die Kunden in zwei Gruppen zu segmentieren. Nutzen Sie dabei die Attribute Alter und Einkommen:

(i) Wählen Sie als Startzentroiden die Werte $Z_1(-0{,}8\ /\ -1{,}1), Z_2(1{,}2/\ 1{,}6)$.

(ii) Berechnen Sie die Zugehörigkeit der ersten **fünf** Datenpunkte zu den Startzentroiden. (Verwenden Sie als Abstandsmaß die euklidische Distanz. Zwei Punkte $P_1(x_1/x_2), P_2(y_1/y_2)$ haben danach den Abstand

$$a = \sqrt[2]{(x_1 - y_1)^2 + (x_2 - y_2)^2}$$

(iii) Aktualisieren Sie die Position der Zentroiden basierend auf den zugewiesenen Datenpunkten.

*Hinweis:* Verwenden Sie für die Berechnung die standardisierten Werte aus folgender Tabelle:

| **Datenpunkt** | **Alter (standardisiert)** | **Einkommen (standardisiert)** |
|---|---|---|
| 1 | -0,8 | -1,1 |
| 2 | 1,2 | 1,6 |
| 3 | -0,9 | -0,3 |
| 4 | -0,6 | -0,6 |
| 5 | -0,8 | -1 |

**Lösungsvorschlag:**

(a) Die Kundendaten umfassen verschiedene Datentypen und Merkmalsarten, die für die Segmentierung und Entwicklung neuer Versicherungsprodukte relevant sind:

**Kategoriale Daten:** Geschlecht, Politische Präferenz, Fahrzeugtyp, Energieverbrauch, Einkaufsgewohnheiten, Teilnahme an Umweltinitiativen (Boolean).

**Skalare Daten:** Alter, Einkommen.

(b) Risiken bei diesem Produkt:

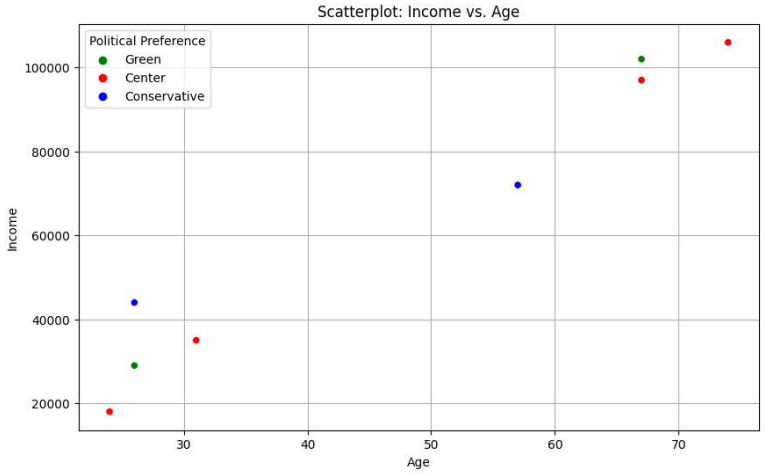
(i) Datenschutzrisiken: Erhöhte Sensibilität bei der Verarbeitung und Speicherung von umweltbezogenen Kundendaten, insbesondere in Bezug auf die DSGVO.

(ii) Marktrisiken: Mögliche Ablehnung oder geringere Akzeptanz der neuen nachhaltigen Produkte bei konservativen oder preissensitiven Kundengruppen.

(iii) Operative Risiken: Herausforderungen bei der Integration neuer Daten und Produktionsprozesse, die die Effizienz und Rentabilität der neuen Produkte beeinträchtigen könnten.

(iv) Reputationsrisiko: Die Einteilung nach politischen Präferenzen kann zu einem Reputationsrisiko führen, insbesondere wenn Kunden das Gefühl haben könnten, dass ihre persönlichen Überzeugungen nicht respektiert oder in unangemessener Weise verwendet werden. Dies könnte zu einem Imageverlust für das Unternehmen führen, wenn die Segmentierung nicht transparent und gerecht kommuniziert wird.

(c) ScatterPlot:



Vermutlich 2 Cluster.

(d) Veränderung des Berufs des Aktuars:

(i) Aktuare müssen verstärkt Datenanalysen durchführen, um Kundenpräferenzen zu verstehen und segmentierte Produkte zu entwickeln.

(ii) Zusammenarbeit mit anderen Abteilungen wie Marketing und IT zur Integration von Daten und zur Entwicklung neuer Produkte.

(iii) Erhöhte Verantwortung im Umgang mit sensiblen Kundendaten und Einhaltung datenschutzrechtlicher Vorschriften

(iv) Erhöhte Anforderungen an das IT-Wissen von Aktuaren zur Erstellung von Datenanalysen mittels neuer Algorithmen und größerer Datenmengen.

(v) …

(e) Berechnungsschritte

Zuerst einmal berechnen wir den Abstand von jedem der fünf Punkte zu den beiden Startzentroiden $Z_1(-0{,}8\ /\ -1{,}1), Z_2(1{,}2/\ 1{,}6)$:

| Datenpunkt | Alter (standardisiert) | Einkommen (standardisiert) | Abstand zu $Z_1$ | Abstand zu $Z_2$ | Cluster |
|---|---|---|---|---|---|
| **1** | -0,8 | -1,1 | 0,00 | 3,36 | 1 |
| **2** | 1,2 | 1,6 | 3,36 | 0,00 | 2 |
| **3** | -0,9 | -0,3 | 0,81 | 2,83 | 1 |
| **4** | -0,6 | -0,6 | 0,54 | 2,84 | 1 |
| **5** | -0,8 | -1 | 0,10 | 3,28 | 1 |

Offensichtlich gehören alle drei weiteren Punkte zur Gruppe um den ersten Zentroiden.

Anschließend berechnen wir die neuen Startzentroiden. Da in Gruppe 2 nur ein Punkt enthalten ist, bleibt dieser Zentroid unverändert.

$$Z_2^{(2)} = Z_2(1{,}2\ /\ 1{,}6)$$

In Gruppe 1 sind vier Punkte enthalten. Damit ergibt sich der neue Zentroid zu:

$$Z_1^{(2)} = \left(\frac{-0{,}8 - 0{,}9 - 0{,}6 - 0{,}8}{4} / \frac{-1{,}1 - 0{,}3 - 0{,}6 - 1}{4}\right) = (-0{,}775/-0{,}75)$$

**Aufgabe 2**. *[1.1 Grundlagen, 3.1 Data Mining 1, 4.2 Unüberwachtes Lernen 1, 5.2 Sprachübergreifende Data Science-Tools] [18 Punkte]*

Ihr Unternehmen verarbeitet Kundendaten aus verschiedenen Quellen und muss daher diverse Datenformate unterstützen. Dabei sind sowohl technische als auch datenschutzrechtliche Anforderungen zu berücksichtigen.

(a) *[6 Punkte]* Ihr Unternehmen muss verschiedene Datenformate unterstützen, um Kundendaten von anderen Unternehmen zu importieren und zu exportieren. Nennen Sie zwei (außer JSON) offene Datenformate, welche die Schnittstelle verarbeiten können soll, und geben Sie jeweils einen Beispieldatensatz für jedes Format an.

(b) *[6 Punkte]* Die Verarbeitung und Speicherung von Kundendaten muss im Einklang mit der Datenschutzgrundverordnung (DSGVO) erfolgen. Erläutern Sie, inwiefern das Speichern von Daten in offenen Formaten Ihrem Unternehmen gemäß Artikel 15 DSGVO hilft, und was dieser Artikel beinhaltet.

(c) *[6 Punkte]* Nennen und beschreiben Sie 3 Tools, die sowohl über eine Benutzeroberfläche als auch mittels R- und/oder Pythonschnittstelle angewendet werden können, um die voraussichtlich sehr große Datenmenge geeignet verarbeiten und analysieren zu können. Benennen Sie jeweils Benutzeroberfläche und Schnittstellen.

**Lösungsvorschlag:**

(a) Offene Datenformate:

(i) XML (Extensible Markup Language):

Beschreibung: XML ist eine Markup-Sprache, die sowohl für die menschliche als auch die maschinelle Lesbarkeit entwickelt wurde und häufig für den Datenaustausch im Web verwendet wird.

Beispiel:

```
<Kunde>
  <Kundennummer>12345</Kundennummer>
  <Name>Max Mustermann</Name>
  <Alter>30</Alter>
  <Versicherungstyp>Kfz-Versicherung</Versicherungstyp>
  <Prämie>450.50</Prämie>
</Kunde>
```

(ii) CSV (CommaSeparated Values):

Beschreibung: CSV ist ein einfaches Format zum Speichern tabellarischer Daten, bei dem jede Zeile ein Datensatz ist und die einzelnen Datenfelder durch Kommata getrennt sind.

Beispiel:

```
Kundennummer,Name,Alter,Versicherungstyp,Prämie
12345,Max Mustermann,30,Kfz-Versicherung,450.50
```

(iii) Andere Beispiele wären .txt, .html, .odt, .gpx, … Dateien

(b) Artikel 15 DSGVO:

Inhalt: Artikel 15 gewährt betroffenen Personen das Recht auf Auskunft über die personenbezogenen Daten, die ein Unternehmen von ihnen speichert. Dies um-

fasst Informationen über den Zweck der Verarbeitung, die Kategorien der verarbeiteten Daten und die Empfänger der Daten.

Vorteile offener Formate:

- Einfacher Datenzugriff: Offene Formate ermöglichen eine leichtere Bereitstellung der angeforderten Informationen, da sie standardisiert und weit verbreitet sind.
- Interoperabilität: Daten in offenen Formaten können leicht zwischen verschiedenen Systemen ausgetauscht werden, was die Erfüllung von Auskunftsanfragen beschleunigt.
- Transparenz: Durch die Nutzung offener Formate kann das Unternehmen sicherstellen, dass die bereitgestellten Daten für die betroffenen Personen verständlich sind.
- Die Bereitstellung in einem gängigen elektronischen Format ist damit sehr einfach.

(c) Mögliche Tools:

**Tableau:**

Beschreibung: Ein leistungsstarkes Tool zur Datenvisualisierung, das komplexe Datenanalysen ermöglicht.

Benutzeroberfläche: Intuitive Drag-and-Drop-Oberfläche.

APIs: Tableau-R und tableau-api-lib für Python.

**Jupyter Notebooks:**

Beschreibung: Eine Open-Source-Webanwendung, die es ermöglicht, Dokumente zu erstellen und zu teilen, die Live-Code, Gleichungen, Visualisierungen und narrative Texte enthalten.

Benutzeroberfläche: Webbasierte Benutzeroberfläche für interaktive Programmierung.

APIs: Direkte Integration mit Python (Jupyter) und R (IRkernel).

**RStudio:**

Beschreibung: Eine integrierte Entwicklungsumgebung für R und Python, die sowohl für statistische Analysen als auch für die Entwicklung von Data Science-Anwendungen verwendet wird.

Benutzeroberfläche: Umfangreiche IDE für R-Programmierung.

APIs: Direkte Integration mit R-Paketen und R-Skripten.

**Power BI:**

Beschreibung: Microsofts Business Analytics-Tool, das detaillierte Einblicke und Visualisierungen ermöglicht.

Benutzeroberfläche: Benutzerfreundliche Drag-and-Drop-Oberfläche für die Erstellung interaktiver Berichte und Dashboards.

APIs: Unterstützung für R und Python direkt in Power BI-Desktop durch integrierte Skript-Visualisierungen.

**KNIME:**

Beschreibung: Eine Open-Source-Software für Datenanalyse, Reporting und Integration, die datenwissenschaftliche Workflows vereinfacht.

Benutzeroberfläche: Visuelle Workflow-basierte Benutzeroberfläche, die es ermöglicht, komplexe Datenprozesse ohne umfangreiche Programmierung zu erstellen.

APIs: Eingebundene Knoten für R und Python, die es ermöglichen, Skripte direkt innerhalb von Workflows auszuführen.

Weitere Toolnennungen sind möglich.

**Aufgabe 3.** *[1.4 Datenschutz 1, 2.3 Informationsverarbeitung in Versicherungen, 3.1 Data Mining 1] [14 Punkte]*

Sie sind Mitglied eines neuen Projektteams, das im Versicherungsunternehmen die Anwendung von maschinellem Lernen erkunden und umsetzen soll. Das Team hat die Aufgabe, erste Konzepte zu entwickeln, die sowohl die Datenqualität verbessern als auch die Grundlagen des maschinellen Lernens in den Unternehmensprozessen einsetzen. Darüber hinaus soll sichergestellt werden, dass alle regulatorischen Anforderungen im Zusammenhang mit dem Einsatz von Künstlicher Intelligenz (KI) in der Versicherungsbranche beachtet werden.

(a) *[6 Punkte]* Das Team hat festgestellt, dass in den bestehenden Kundendaten einige Ungenauigkeiten und Lücken vorhanden sind. Benennen Sie drei konkrete Maßnahmen, um die Datenqualität zu verbessern, und erklären Sie, wie Sie diese Maßnahmen in einem Versicherungsunternehmen anwenden würden.

(b) *[5 Punkte]* Erklären Sie die Unterschiede zwischen überwachtem und unüberwachtem Lernen im Kontext maschineller Lernverfahren. Nennen Sie jeweils ein konkretes Anwendungsbeispiel für überwachtes und unüberwachtes Lernen in der Versicherungsbranche.

(c) *[3 Punkte]* Nennen Sie die zwei grundlegenden Rechtsquellen, die beim Einsatz von Datenanalysen und künstlicher Intelligenz beachtet werden müssen.

**Lösungsvorschlag:**

(a) Maßnahmen zur Steigerung der Datenqualität:

**Präventive Maßnahmen:**

- Vermeidung der fehlerhaften Erfassung von Daten durch
  - Verwendung von übersichtlichen Auswahllisten
  - Automatische Überprüfung von Eingaben
  - Unwissen zulassen / abgestufte Plausibilisierung
  - Zeitpunkt der Datenerhebung festlegen
- Einführung Datenarchitektur / unternehmensweites Datenmodell
  - Zuordnung der Daten zu Geschäftsdomänen
  - Vermeidung von Redundanz
  - Verwendung von fachlichen Definitionen
  - Semantische Überlagerungen auflösen

**Maßnahmen auf gegebenen Daten:**

- Messen der Datenqualität durch Abgleich gegen Kenngrößen (Ausreißer etc. finden)
- Regelbasiert auf Basis definierter Abweichungstoleranzen

(b) **Überwachtes Lernen**

Beim überwachten Lernen wird ein Modell mithilfe von Daten trainiert, die bereits markiert oder klassifiziert sind (d.h. es gibt Eingabe- und Zielvariablen). Das Ziel ist es, auf Basis der Eingabedaten Vorhersagen zu treffen oder Klassifikationen vorzunehmen.

Beispiel in der Versicherungsbranche: Ein Modell zur Vorhersage von Schadensfällen in der Autoversicherung. Anhand von historischen Daten (z.B. Fahrzeugalter, Fahrverhalten, etc.) und den dazugehörigen Schadensfällen (Zielvariablen) kann das Modell trainiert werden, zukünftige Schäden vorherzusagen.

**Unüberwachtes Lernen**

Beim unüberwachten Lernen gibt es keine vorgegebenen Zielvariablen. Das Modell wird darauf trainiert, Muster und Strukturen in den Daten zu erkennen.

Beispiel in der Versicherungsbranche: Segmentierung von Kunden nach Ähnlichkeiten im Verhalten. Ein Versicherungsunternehmen könnte beispielsweise unüberwachtes Lernen verwenden, um Kundengruppen mit ähnlichen Eigenschaften (z.B. Alter, Risikoverhalten, Prämienhöhe) zu identifizieren, die dann für gezielte Marketingmaßnahmen genutzt werden könnten.

(c) **1. DSGVO** (Datenschutz-Grundverordnung)

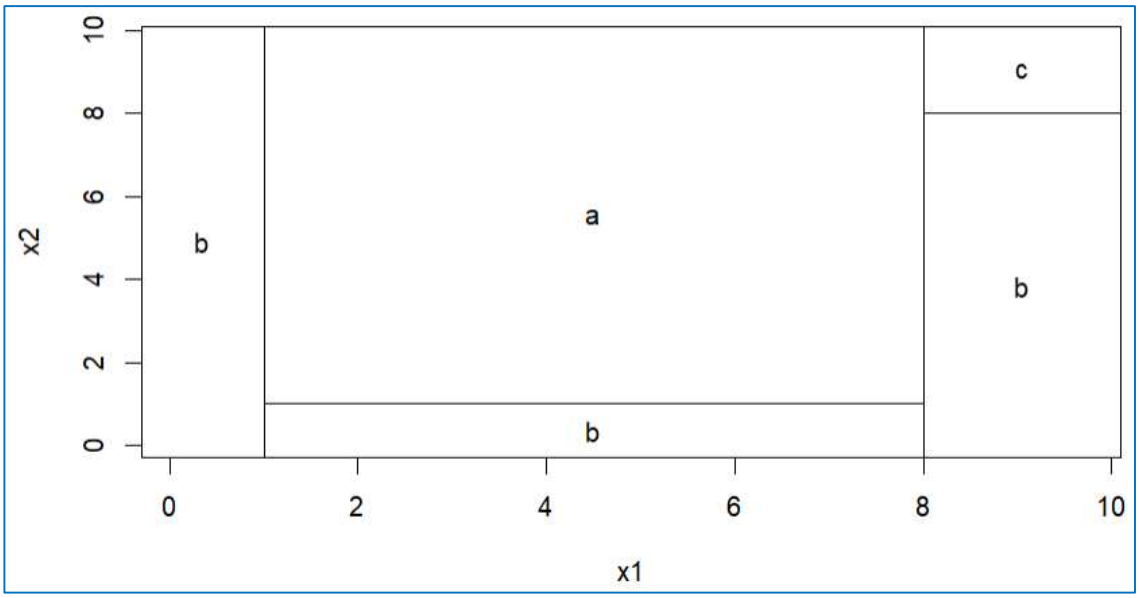
Die DSGVO regelt den Schutz personenbezogener Daten in der Europäischen Union. Beim Einsatz von KI und Datenanalysen in der Versicherungsbranche ist es entscheidend, sicherzustellen, dass die Verarbeitung von Kundendaten den Anforderungen der DSGVO entspricht, insbesondere was die Einwilligung zur Datennutzung, das Recht auf Vergessenwerden und den Schutz sensibler Daten betrifft.

**2. EU AI Act** (Europäische Verordnung zur Regulierung Künstlicher Intelligenz)

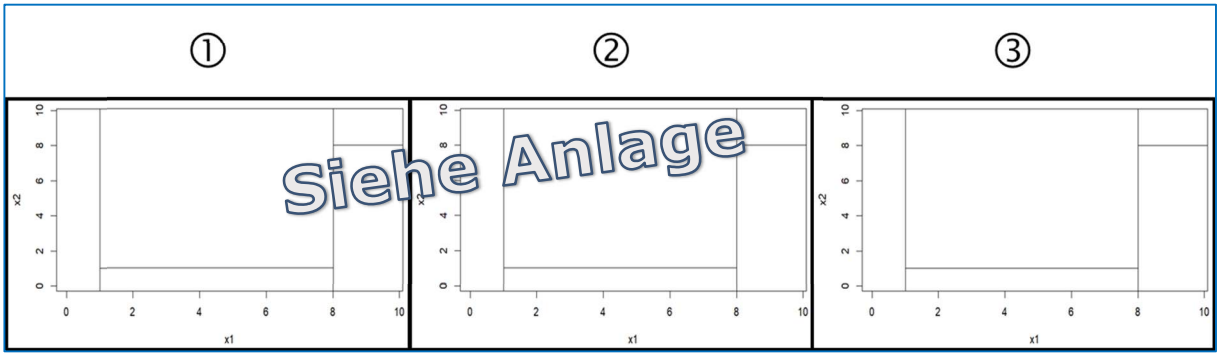
Der EU AI Act ist ein regulatorischer Rahmen, der darauf abzielt, die Entwicklung und den Einsatz von KI-Systemen in der EU zu regeln. Er unterscheidet zwischen verschiedenen Risikoklassen von KI-Anwendungen, die von minimalem bis hin zu inakzeptablem Risiko reichen. Versicherungsunternehmen, die KI einsetzen, müssen sicherstellen, dass ihre Systeme transparent, sicher und fair sind und die gesetzlichen Vorgaben einhalten.

**Aufgabe 4.** *[4.1 Überwachtes Lernen 1] [27 Punkte]*

(a) *[5 Punkte]* Im Folgenden sehen Sie eine Partition eines 2-dimensionalen Merkmalsraums, die durch Klassifikation mittels CART-Algorithmus entstanden ist. Zeichnen Sie einen hierzu passenden Entscheidungsbaum. Runden Sie dabei numerische Werte aus der untenstehenden Grafik auf ganzzahlige Werte.




(b) *[6 Punkte]* Hinweis: Benutzen Sie zur Beantwortung von Teilaufgabe b) bitte das der Klausur beigefügte Blatt mit drei Grafiken ① bis ③. Verwenden Sie für Ihre Ergänzungen ein Lineal.




Ausgehend von der Partitionsgrafik aus Teilaufgabe a): Zeichnen Sie gerade Linien ein, die wieder zu einer Partition führen, mit unterschiedlichen Nebenbedingungen.

① Zeichnen Sie eine gerade Linie ein, sodass die neue Partition ebenfalls durch den CART-Algorithmus entstanden sein könnte.

② Zeichnen Sie eine einzige gerade Linie ein, sodass die neue Partition keinen Entscheidungsbaum abbildet.

③ Zeichnen Sie alternativ zwei gerade Linien ein, durch die sich eine substantiell andere Partition ergibt, die ebenfalls keinen Entscheidungsbaum abbildet.

Erläutern Sie Ihre Entscheidungen.

Achten Sie darauf, die von Ihnen bearbeitete Seite zusätzlich abzugeben!

(c) *[7 Punkte]* Nennen Sie 3 Vor- und 2 Nachteile von Entscheidungsbäumen gegenüber anderen ML-Verfahren.

Beschreiben Sie in zwei bis drei Sätzen einen Ansatz, durch den man einen Entscheidungsbaum erhält, bei dem einer der von Ihnen genannten Nachteile behoben ist.

(d) *[9 Punkte]* Was ist (in ein bis zwei Sätzen) das gemeinsame Prinzip aller sog. Ensemble-Methoden?

Erläutern Sie in jeweils einem Satz die Idee von Bagging bzw. Boosting – unabhängig von Entscheidungsbäumen.

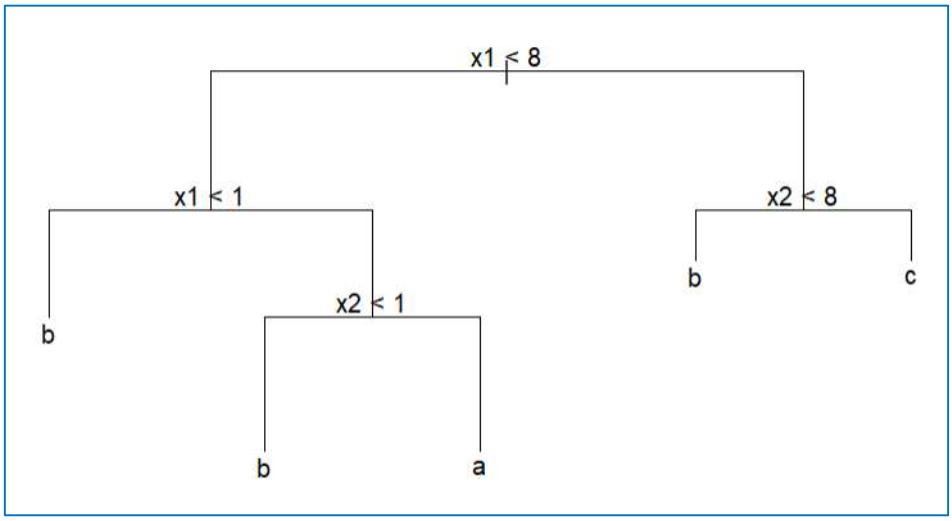
Inwiefern lassen sich durch Bagging oder Boosting gewisse Nachteile von Entscheidungsbäumen mindern?

Wie passen Random Forests zu diesen Konzepten und was ist die Besonderheit von Random Forests und warum?

Welcher wesentliche Vorteil von Entscheidungsbäumen geht beim Random Forest verloren?

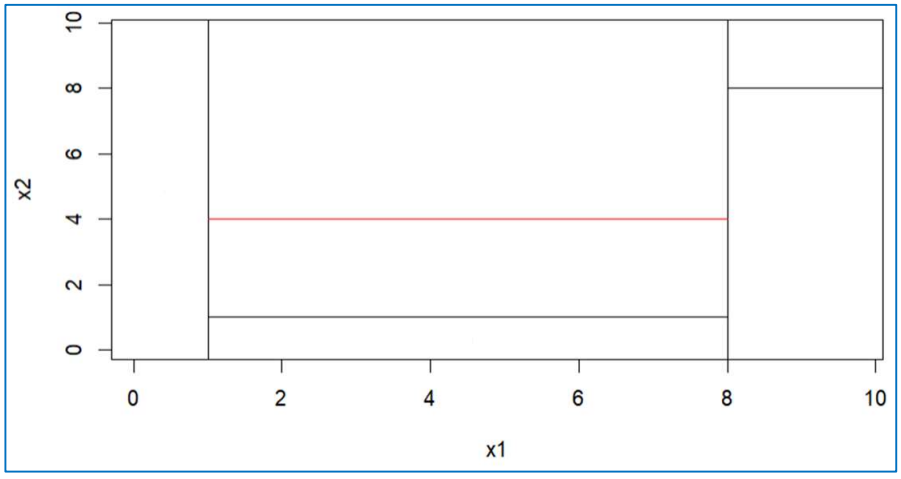
**Lösungsvorschlag:**

(a) Der folgende Baum zeigt jeweils die Prüfung (hier mit „<“; „≤“ wäre ebenso möglich) und verzweigt dann nach links im Fall „ja“, nach rechts im Fall „nein“. In den Blättern wird die jeweilige Klassifikation entsprechend der Partitionsgrafik angegeben.

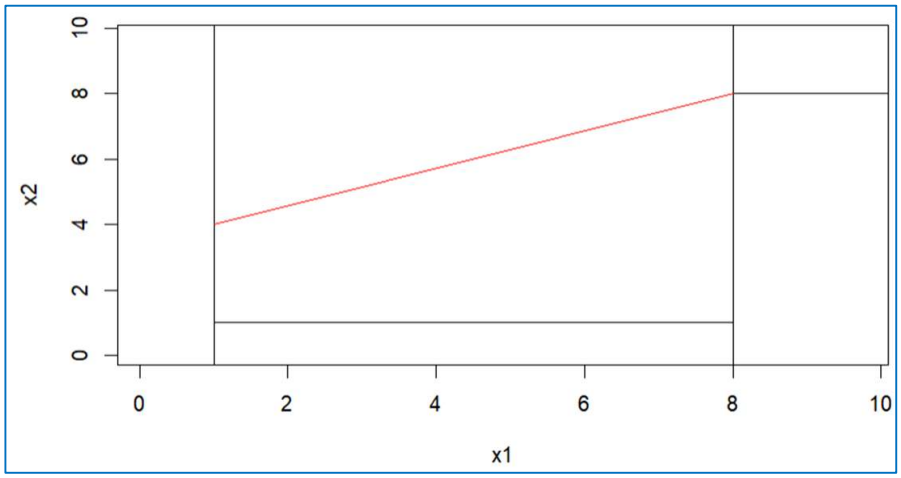



*Anm.: Andere Lösungen sind möglich; je nachdem wie die (hier nicht bekannten) Originaldaten verteilt sind, kann zunächst (d.h. ganz oben im Baum) die Prüfung „x1<1“ erfolgen und die Prüfung „x1<8“ erst auf der Ebene darunter.*

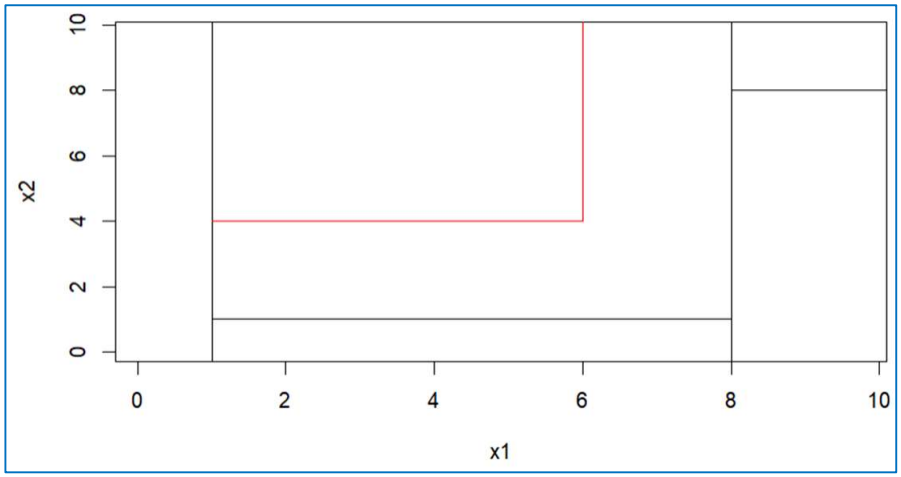
(b) ① Eine zusätzliche achsenparallele Linie, die an die bereits vorhandenen Linien anschließt, führt zu einer Partition, die ebenfalls durch den CART-Algorithmus entstanden sein könnte:




② Eine zusätzliche nicht achsenparallele Linie kann niemals durch den CART-Algorithmus entstehen:




③ Ein durch zwei zusätzliche achsenparallele Linien gebildeter Bereich kann nicht zu einer CART-bedingten Partition gehören, wenn diese beiden Linien nicht bis zum Rand bzw. bis zu bereits vorhandenen Linien reichen:




(c) **Vorteile von Entscheidungsbäumen:** *(es genügt die Nennung von 3 Vorteilen)*

1. **Einfache Interpretierbarkeit:** Entscheidungsbäume sind leicht zu verstehen und zu interpretieren und damit auch zu kommunizieren. Insbesondere kleinere, nicht zu tiefe Bäume lassen sich vergleichsweise einfach visualisieren. Jeder Knoten stellt eine Entscheidung dar, die klar nachvollziehbar ist, was sie besonders für den Einsatz in Bereichen mit strengen regulatorischen Anforderungen attraktiv macht.

2. **Skalierung der Daten überflüssig:** Entscheidungsbäume benötigen keine Daten-Normalisierung oder -Standardisierung. Sie können direkt mit kategorischen und numerischen Daten arbeiten, ohne dass Vorverarbeitungsschritte wie die Skalierung notwendig sind.

3. **Umgang mit fehlenden Werten:** Entscheidungsbäume können fehlende Daten verarbeiten, indem sie für fehlende Attribute Surrogate-Splits verwenden oder fehlende Werte während des Trainings ignorieren.

4. **Robustheit gegenüber Ausreißern:** Entscheidungsbäume sind wenig anfällig für Ausreißer, da sie Daten in diskrete Segmente unterteilen und nicht durch extreme Werte verzerrt werden.

5. **Identifikation aussagekräftiger Merkmale:** Entscheidungsbäume eignen sich außerdem gut, um die aussagekräftigsten Features zu identifizieren.

**Nachteile von Entscheidungsbäumen:** *(es genügt die Nennung von 2 Nachteilen)*

1. **Overfitting:** Entscheidungsbäume neigen dazu, sich stark an die Trainingsdaten anzupassen, was zu einer schlechten Generalisierung auf neue Daten führen kann.

2. **Instabilität:** Kleine Änderungen in den Trainingsdaten können zu erheblichen Änderungen in der Struktur des Entscheidungsbaums führen, was sich negativ auf die Interpretierbarkeit und Verlässlichkeit der Ergebnisse auswirkt.

3. **Keine komplexen Zusammenhänge:** Entscheidungsbäume haben möglicherweise Schwierigkeiten, komplexe Beziehungen in den Daten zu erfassen, was in bestimmten Szenarien zu Leistungseinbußen führt.

*Anm.: Für die volle Punktzahl bei den Vor- und Nachteilen genügt es, jeweils drei bzw. zwei klar verständliche Stichworte zu nennen (entsprechend den fettgedruckten Begriffen in der obigen Auflistung).*

Das Problem des Overfittings kann innerhalb des Bereichs der Entscheidungsbäume durch die Technik des Beschneidens (Pruning) abgemildert werden. Hierdurch wird die Komplexität eines Entscheidungsbaums reduziert. Das Beschneiden kann vorausschauend (durch geeignete Stoppkriterien) oder nachträglich (durch Vergleich mehrerer Bäume z.B. via Kreuzvalidierung) erfolgen.

Anm.: Die Verwendung von Ensemble-Methoden wie Random Forests führt zwar ebenfalls zu einer Reduktion von Overfitting, verlässt aber den Bereich der Entscheidungsbäume im engeren Sinne.

(d) **Ensemble-Methoden:**

Ensemble-Methoden kombinieren die Vorhersagen mehrerer ML-Modelle, um die Gesamtgenauigkeit zu verbessern und robustere und leistungsfähigere Vorhersagen zu erhalten als bei der Nutzung eines einzelnen Modells.

Bagging und Boosting sind Beispiele für solche Ensemble-Methoden.

**Bagging** (Bootstrap Aggregating) kombiniert mehrere Modelle, die auf unterschiedlichen Stichproben der Trainingsdaten trainiert wurden, um deren Vorhersagen zu mitteln (durch Durchschnittsbildung oder Mehrheitswahl) und dadurch die Varianz zu reduzieren und die Stabilität des Modells zu erhöhen.

**Boosting** ist eine iterative Methode, bei der schwache Modelle (Modelle mit geringerer Vorhersagegenauigkeit) nacheinander trainiert werden, wobei jedes neue Modell die Fehler der vorherigen Modelle korrigiert, um so schrittweise ein starkes Gesamtmodell zu bilden.

Die erwähnte Varianzreduktion beim Bagging geht einher mit verringerter Neigung zum Overfitting und wirkt damit diesem Nachteil von Entscheidungsbäumen entgegen. Boosting hingegen steigert die Modellgenauigkeit durch sukzessive Fehlerkorrektur, wodurch sowohl der Bias reduziert als auch die Generalisierungsfähigkeit verbessert wird.

Ein **Random Forest** ist ein Ensemble von Entscheidungsbäumen, basierend auf dem Bagging-Ansatz. Allerdings sind die so gebootstrappten Vorhersagen i.d.R. nicht voneinander unabhängig, weil dominante Features zu ähnlichen Splits und damit ähnlichen Bäumen führen. Daher wird beim Random Forest in einem zweiten Schritt eine Random Subset Selection vorgenommen, d.h. es wird bei jedem Split eine zufällige Menge von Features ausgewählt.

Der **Nachteil von Random Forests** gegenüber Entscheidungsbäumen ist der Verlust der Interpretierbarkeit.

**Aufgabe 5.** *[5.1 Programmiersprachen für Data Science] [18 Punkte]*

(a) *[5 Punkte]* In dieser Teilaufgabe benutzen wir das Dataset women aus dem vorinstallierten R-package datasets. Hierzu finden Sie in der R-Dokumentation insbesondere folgende Informationen:

women {datasets} R Documentation

**Average Heights and Weights for American Women**

**Description**

This data set gives the average heights and weights for American women aged 30–39.

**Usage**

```
women
```

**Format**

A data frame with 15 observations on 2 variables.

`[,1]` `height` numeric Height (in)

`[,2]` `weight` numeric Weight (lbs)

Schreiben Sie gut lesbaren R-Code unter maximaler Nutzung von `tidyverse` (d.h konkret insbesondere `dplyr` und des Pipe-Operators), um folgende Aufgabenstellung zu erfüllen. Führen Sie dabei alle Schritte der Reihe nach aus und überspringen Sie keinen und beachten Sie auch die Hinweise, die im Anschluss gegeben sind:

> Laden Sie `tidyverse`; berechnen Sie die Variable `X1`, indem Sie ausgehend von `women` in einer neuen Spalte `bmi` den Body-Mass-Index (BMI) berechnen, die Tabelle auf Zeilen mit BMI≥23 einschränken und die Anzahl dieser Zeilen bestimmen (das sollte der Inhalt der Variable `X1` sein).

*Hinweise:*
BMI = “Körpergewicht in Kilogramm geteilt durch die Körpergröße in Metern zum Quadrat”.
Umrechnungsfaktor Zoll (in) → Meter: 0,0254.
Umrechnungsfaktor Amerikanische Pfund (lbs) → Kilogramm: 0,45359237.

(b) *[7 Punkte]* Wie sieht die Ausgabe aus, die durch folgenden R-Code erzeugt wird? Benutzen Sie für die Grafik ein Lineal.

Für Grafikeigenschaften, die durch den Code nicht explizit festgelegt sind, treffen Sie bitte geeignete sinnvolle Annahmen.

```
c <- tibble(sex=c("m","m","w","w","w"),
       ins=c(TRUE, FALSE, FALSE, TRUE, FALSE),
       weight=c(75,95,75,90,95),
       height=c(1.8,1.6,1.6,1.75,1.8))

c

c %>% ggplot(aes(x=height, y=weight, shape=factor(sex))) +
      geom_point(size=3) +
      labs(title="What does this plot show?", shape="sex")
```

Wie verändert sich die Grafik, wenn Sie stattdessen folgenden Befehl eingeben:

```
c %>% ggplot(aes(x=height, y=weight, shape=factor(sex), size=3)) +
      geom_point() +
      labs(title="What does this plot show?", shape="sex")
```

Warum ergibt sich dieses (unerwünschte) Verhalten?

(c) *[6 Punkte]* Erläutern Sie die Begriffe Copy-on-Modify und Modify-in-Place jeweils an einem kurzen R-Code-Beispiel. Welche Rolle spielen diese Konzepte in R?

**Lösungsvorschlag:**

(a) Code für die Ausführung der Aufgabenstellung:

```
library(tidyverse)

X1 <- women %>%
      mutate(bmi=(0.45359237*weight)/(0.0254*height)^2) %>%
      filter(bmi>=23) %>%
      nrow
```

Anm.: Die in der Aufgabenstellung zitierten Informationen aus der R-Dokumentation finden sich in dieser Form unter https://search.r-project.org/R/refmans/datasets/html/women.html.

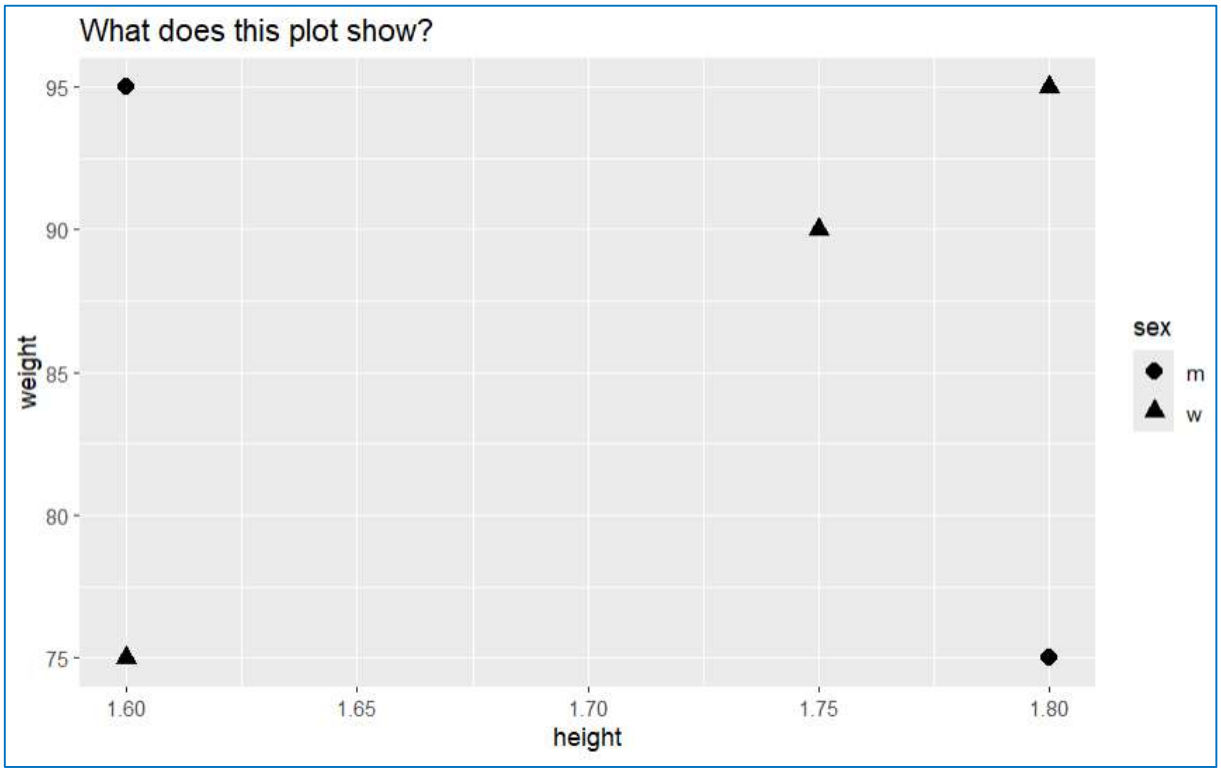
(b) Der Aufruf von c liefert folgende **tabellarische Darstellung:**

A tibble: 5 × 4

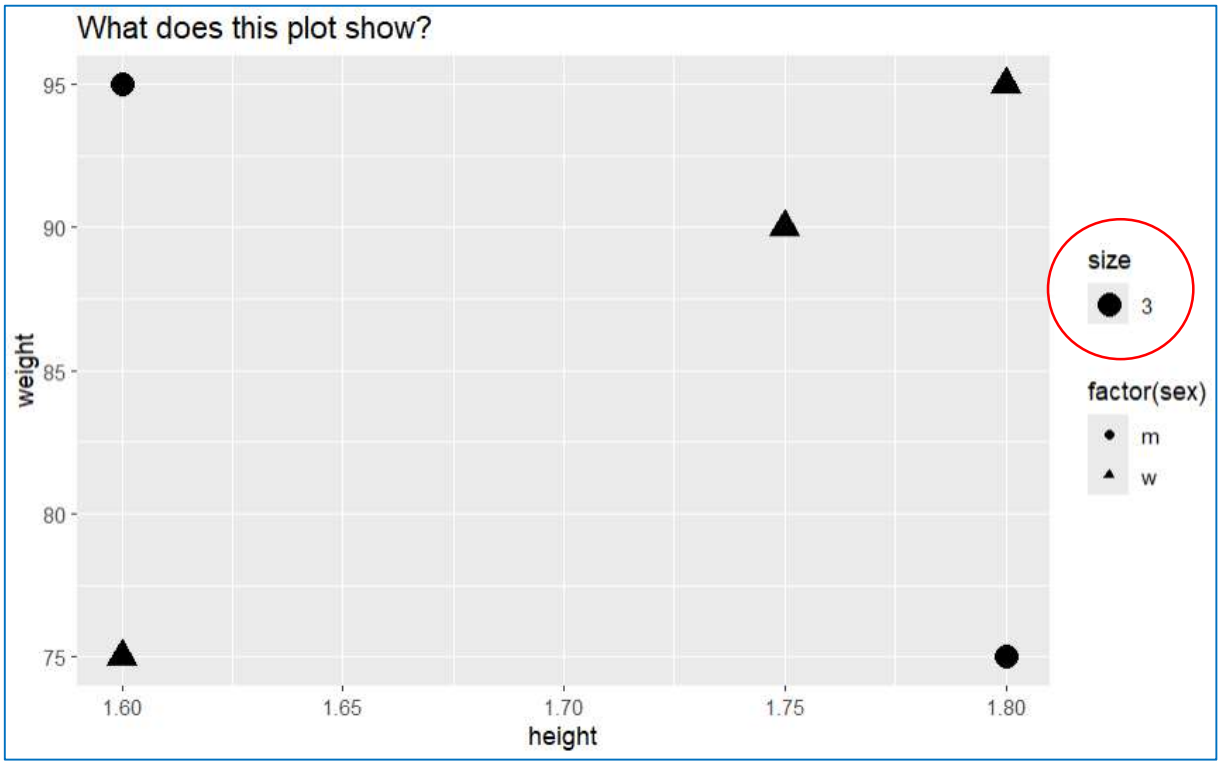
| sex <chr> | ins <lgl> | weight <dbl> | height <dbl> |
|---|---|---|---|
| m | TRUE | 75 | 1.80 |
| m | FALSE | 95 | 1.60 |
| w | FALSE | 75 | 1.60 |
| w | TRUE | 90 | 1.75 |
| w | FALSE | 95 | 1.80 |

5 rows

Die **Grafikausgabe** sieht folgendermaßen aus:




Der **alternative `ggplot`-Aufruf** liefert dagegen eine zusätzliche Legende für eine Variable `size` (in der folgenden Beispielgrafik durch einen roten Kreis hervorgehoben). Das liegt daran, dass in diesem Fall die Zuweisung „`size=3`“ fälschlicherweise innerhalb der `aes`-Funktion eingebunden wird, also als ästhetisches Mapping für eine konstante „Variable“ 3 interpretiert wird, die über die Größe (`size`) der Datenpunkte oder Linien wiedergegeben werden soll; die Größe der „Punkte“ wird durch diesen Wert 3 nicht beeinflusst (z.B. erhält man mit dem Wert 10 die gleiche Grafik – von der 10 in der Legende abgesehen). Im korrekten `ggplot`-Aufruf wird `size` dagegen als Argument für `geom_point()` verstanden und beeinflusst die Größe der „Punkte“.




(c) Modify-in-Place und Copy-on-Modify sind Konzepte, die in R zum Management des allokierten Speicherplatzes beim Modifizieren von Objekten verwendet werden. Die beiden Verfahren unterscheiden sich hinsichtlich der Sparsamkeit des Speicherbedarfs.

- Bei **Modify-in-Place** wird ein bereits in R existierendes Objekt durch die Modifikation von dessen zugewiesenem Wert im Speicher verändert. Es wird keine Kopie angelegt und der alte Stand ist nicht mehr verfügbar. Je nach Speicherbedarf des neu gebundenen Werts wird ggf. mehr Speicher in Anspruch genommen bzw. wieder freigegeben.

- Bei **Copy-on-Modify** wird durch die Modifikation des Werts eines Objekts eine Kopie des Ausgangsobjekts im Speicher angelegt, d.h. das bisherige Objekt existiert in seiner bisherigen Form weiterhin im Speicher. Prinzipiell wird hierdurch mehr Speicher benötigt.

Für Objekte, für die nur ein eindeutiges Binding existiert (d.h. nur von einem Namen referenziert werden), wird i.Allg. Modify-in-Place angewandt. Im folgenden Beispiel existiert nach Ausführung des Programms der Wert 3 nicht mehr im Speicher, das Binding x referenziert aber weiterhin auf das Ausgangsobjekt mit modifiziertem Wert 4.

```
x <- c(1, 2, 3)
x[3] <- 4
x

## [1] 1 2 4
```

Modify-in-Place wird außerdem für Umgebungen (Objekte vom Typ `environment`) immer angewandt.

Ansonsten verwendet R stets Copy-on-Modify. Nachdem bspw. ein atomischer Vektor ein zweites Binding erhält, referenzieren zunächst beide Namen auf dieselben Werte im Speicher. Wird nun durch Subassignment auf einen der beiden Namen ein Wert des Objekts modifiziert, so existieren sowohl das Ausgangs- als auch das modifizierte Objekt mit dem jeweiligen Namen als Referenz im Speicher (auch wenn weiterhin viele Elemente der beiden Vektoren gleiche Werte haben). Im folgenden Beispiel liegen nach Programmausführung sowohl ein Objekt mit Werten `c(1, 2, 3)` als auch ein Objekt mit Werten `c(1, 2, 4)` im Speicher.

```
x <- c(1, 2, 3)
y <- x
y[[3]] <- 4
x

## [1] 1 2 3
```

**Aufgabe 6.** *[Datenmanagement 1, Analytics 1] [25 Punkte]*

(a) *[2 Punkte]* Für die Bewertung von zwei konkurrierenden Modellen erhalten Sie die notwendigen Daten in JSON-Dateien. Das Einlesen der Dateien schlägt jedoch fehl. Anbei der Inhalt einer der JSON-Dateien:

```
{
  "model_id": "djkzqod6_2024",
  "model_columns: ["id i", "score s", "prob p", "target t"],
  "model_eval": [
      {"i": 1, "s": 150, "p": 0.05, "a": 0},
      {"i": 2, "s": 250, "p": 0.1, "a": 0},
      {"i": 3, "s": 370, "p": 0,3, "a": 1},
      {"i": 4, "s": 550, "p": 0.7, "a": 0},
      {"i": 5, "s": 650, "p": 0.8, "a": 0},
      {"i": 6, "s": 750, "p": 0.9, "a": 1, "final": True}],
      "validated": 2024-09-27
}
```

Identifizieren Sie Fehler in diesem Inhalt (nach dem JSON-Standard ECMA-404 bzw. RCF 8259).

(b) *[4 Punkte]* Nach den notwendigen Korrekturen sollen die Daten auf einer Festplatte gespeichert werden. Was ist dabei generell zu beachten? Dabei sollen die Informationen als Textdateien gespeichert werden. Gehen Sie u.a. insbesondere auf die Formate ASCII, ANSI und UTF-8 ein.

(c) *[4 Punkte]* Definieren Sie für den Anwendungsfall einer binären Klassifikation die folgenden Begriffe

Konfusionsmatrix, Präzision, Sensitivität und Spezifität.

(d) *[10 Punkte]* Aus dem JSON aus Aufgabe a) extrahieren Sie nach den notwendigen Korrekturen folgende Datensätze (dabei bedeutet ein Target von 1 einen Ausfall nach dem betrachteten Modell):

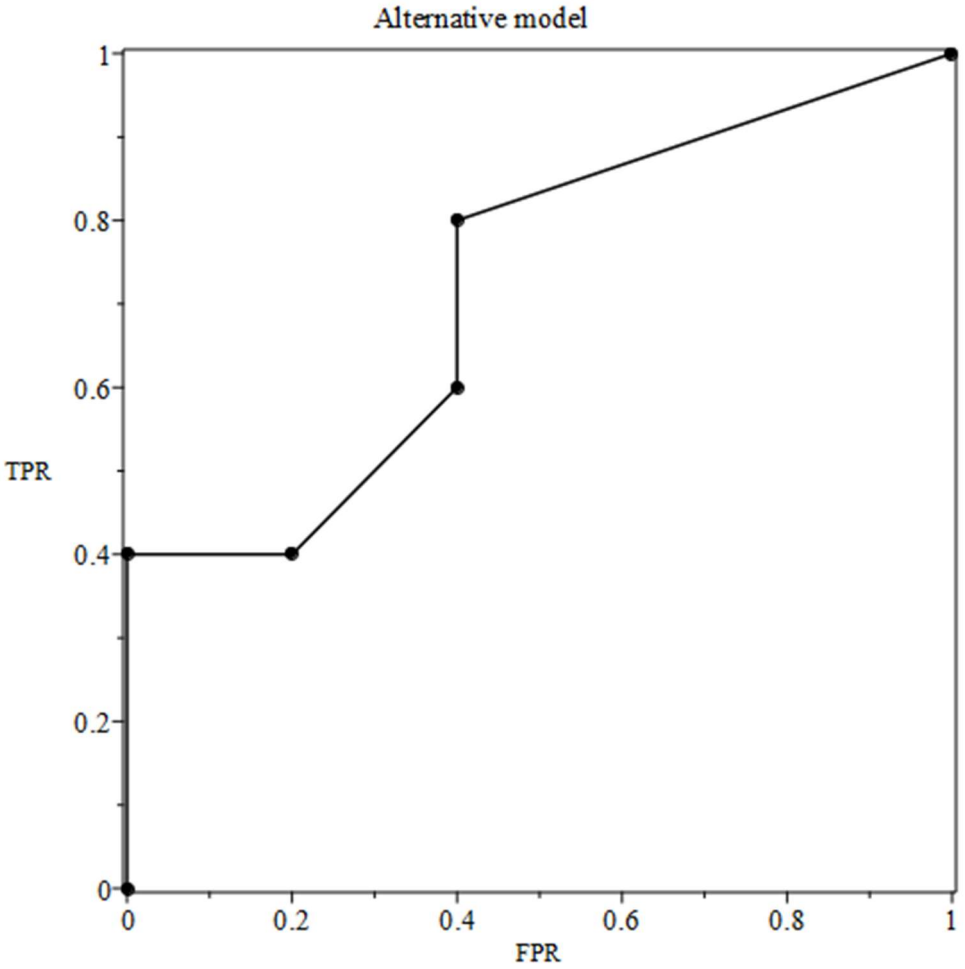
| **ID** | **Score (s)** | **Probability (p)** | **Target (a)** |
|---|---|---|---|
| 1 | 150 | 0.05 | 0 |
| 2 | 250 | 0.1 | 0 |
| 3 | 370 | 0.3 | 1 |
| 4 | 550 | 0.7 | 0 |
| 5 | 650 | 0.8 | 0 |
| 6 | 750 | 0.9 | 1 |

Konstruieren Sie daraus ein entsprechendes ROC-Schaubild mit einer dazugehörigen ROC-Kurve. Berechnen Sie im Anschluss den AUC Wert.

*Hinweis:* Die Rechnungsschritte müssen nachvollziehbar sein. Für einen Wert $p$ und einen Schwellenwert $\alpha > 0$ soll gelten:

$$Klassifizierung = \begin{cases} 0\,, falls\, p \leq \alpha \\ 1, falls\, p > \alpha \end{cases}$$

(e) [5 Punkte] Für ein alternatives Modell haben Ihre Kollegen ein entsprechendes ROC-Diagramm bereits angefertigt:




Berechnen Sie für dieses Modell den AUC-Wert. Vergleichen Sie im Anschluss beide Modelle bzgl. des AUC-Scores. Welches Modell würden Sie bevorzugen? Ist die alleinige Berücksichtigung des AUC-Wertes ausreichend? Nennen Sie zwei weitere Methoden zur Analyse und beschreiben Sie diese.

*Hinweis:* Falls Teil d) nicht bearbeitet wurde, kann für diesen Teil ein AUC-Wert von 0.8 angenommen werden. Dieser Wert stellt nicht das korrekte Ergebnis von Teil d) dar!

**Lösungsvorschlag:**

(a) Nach den angegebenen Standards finden sich in dem JSON folgende Fehler (Dabei steht "{" in der obigen Angabe in Zeile 1):

a. Fehlendes Hochkomma für den String "model_columns" in Zeile 3

b. Fehlender Punkt in Zeile 7 für die Zahl 0.3

c. In Zeile 10 ist das Schlüsselwort true, nicht True

d. Ein Datumstyp ist in JSON nicht vorgesehen. Somit muss das Datum in Zeile 11 als String angegeben werden.

Anm.: Das angepasste JSON kann z.B. über jsonlint.com validiert werden.

(b) Bei der Speicherung auf einem digitalen Medium (z.B. Festplatte) ist insbesondere für textbasierte Dateien die verwendete Codierung zu beachten. Historisch gesehen ist hierbei als Beispiel ASCII zu nennen. Hier besteht die Zeichenkodierung aus insgesamt 128 Zeichen. ANSI ist eine Erweiterung (von 7 auf 8 Bits), womit insgesamt 256 verschiedene Zeichen möglich sind. Der UTF-8 Standard ist heutzutage die am weitesten verbreitete Kodierung für Unicode-Zeichen. Hierbei wird eine unterschiedliche Anzahl an Bytes für die Speicherung von Zeichen verwendet. Die Angabe einer Kodierung ist wichtig, da eine korrekte Verarbeitung nur unter Angabe des richtigen Codierungsformats erfolgen kann.

(c) Definition der Begriffe:

a. Konfusionsmatrix: Im binären Klassifikationsfall stellt diese Matrix die Kennzahlen TP (True Positive), FP (False Positive), FN (False Negative), TN (True Negative) gegenüber. Dabei bedeutet exemplarisch TP, dass eine Beobachtung im Datensatz mit Positiv (=1) Merkmal vom Modell ebenfalls als Positiv (=1) bewertet wurde.

Beispielhaft kann eine Konfusionsmatrix in der folgenden Form angegeben werden:

| | | Predicted (vorhergesagt) | |
|---|---|---|---|
| | | Ja | Nein |
| Actual (tatsächlich) | Ja | TP | FN |
| | Nein | FP | TN |

b. Präzision:

$$Präzision = \frac{TP}{TP + FP}$$

c. Sensitivität (Empfindlichkeit):

$$Sensitivität\ (Recall) = \frac{TP}{TP + FN}$$

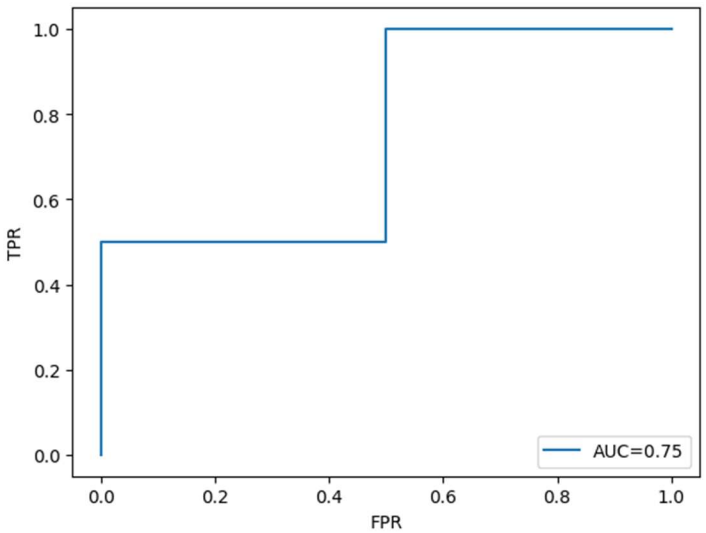
d. Spezifität:

$$Spezifität = \frac{TN}{TN + FP}$$

(d) Die angegebene Tabelle ist bereits nach der Spalte „Probability“ aufsteigend sortiert. Die für die ROC-Kurve benötigten Kennzahlen sind die True Positive Rate (TPR), die gleich ist mit der Sensitivität. Die False Positive Rate (FPT) ist definiert durch 1 – Spezifität. Mit Hilfe der folgenden Schwellenwerte kann die ROC-Kurve konstruiert werden (beispielhaft):

| ID | Schwellenwert | TP | FN | FP | TN | TPR | FPR |
|---|---|---|---|---|---|---|---|
| 1 | 1 | 0 | 2 | 0 | 4 | 0 | 0 |
| 2 | 0.89 | 1 | 1 | 0 | 4 | 0.5 | 0 |
| 3 | 0.79 | 1 | 1 | 1 | 3 | 0.5 | 0.25 |
| 4 | 0.69 | 1 | 1 | 2 | 2 | 0.5 | 0.5 |
| 5 | 0.29 | 2 | 0 | 2 | 2 | 1 | 0.5 |
| 6 | 0.09 | 2 | 0 | 3 | 1 | 1 | 0.75 |
| 7 | 0 | 2 | 0 | 3 | 0 | 1 | 1 |

Daraus ergibt sich folgende ROC-Kurve:




Der AUC-Wert kann daraus geometrisch bestimmt werden:

$$AUC = 0.5 \times 0.5 + 0.5 \times 1 = 0.75$$

(e) Für das alternative Modell kann der AUC-Wert geometrisch bestimmt werden:

$$AUC = 2 \times 0.2 \times 0.4 + 0.6 \times 0.8 + 0.5 \times 0.2 \times 0.2 + 0.5 \times 0.6 \times 0.2 = 0.72$$

Da der Wert für das alternative Modell mit 0.72 unter dem Wert des Modells aus Teil d) mit 0.75 liegt, wäre das Modell aus Teil d) nach AUC zu bevorzugen. Daneben können weitere Kriterien geprüft werden:

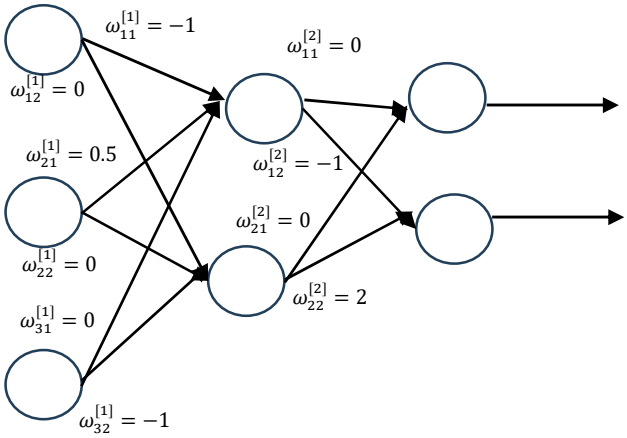
- Kalibrierung: Wie gut stimmen die berechneten Wahrscheinlichkeiten mit den beobachteten Häufigkeiten überein? Hierfür können z.B. Kalibrierungsplots eingesetzt werden.

- Lift-Plots (und darauf basierenden Kennzahlen, z.B. Maximaler Lift Punkt), um das Verhalten der Modelle über den Datensatz besser verstehen zu können und kritische Bereiche zu begutachten.

**Aufgabe 7.** *[Deep Learning 1] [30 Punkte]*

(a) *[7 Punkte]* Erläutern Sie das prinzipielle Vorgehen beim Trainieren eines Feed Forward Neural Networks (FFNN). Fertigen Sie hierzu eine Skizze an um beschreiben Sie wichtige Elemente.

(b) *[8 Punkte]* Gegeben sei das folgende neuronale Netzwerk:




Dabei steht $\omega_{i,j}^{[k]}$ für ein Gewicht von Knoten $i$ in Schicht $k$ zu Knoten $j$ in Schicht $k+1$.

Als Aktivierungsfunktion soll für alle Layer die Sigmoid-Funktion verwendet werden, d.h.

$$\sigma(x) = \frac{1}{1 + e^{-x}}.$$

Geben Sie die benötigten Gewichtsmatrizen an und führen Sie einen Forward-Schritt mit dem Input-Vektor $x = (1, 0, 1)^T$ durch.

(c) *[15 Punkte]* Für eine Backpropagation soll die Lossfunktion

$$L = \left\| a^{[2]} - y \right\|_2^2 = \left( a_1^{[2]} - y_1 \right)^2 + \left( a_2^{[2]} - y_2 \right)^2$$

berücksichtigt werden. Dabei bezeichnet

$$a^{[2]} = \left(a_1^{[2]}, a_2^{[2]}\right)^T$$

die Ausgabe des Output Layers und

$$y = (y_1, y_2)^T$$

den Soll-Wert (die erwartete Ausgabe) des Netzwerks.

Für eine Aktualisierung der Gewichte ist der Gradient der Lossfunktion L von zentraler Bedeutung. Leiten Sie hierfür eine Formel für die partielle Ableitung der Lossfunktion L nach dem Gewicht $\omega_{11}^{[2]}$ (siehe Skizze in b) ) her, d.h.

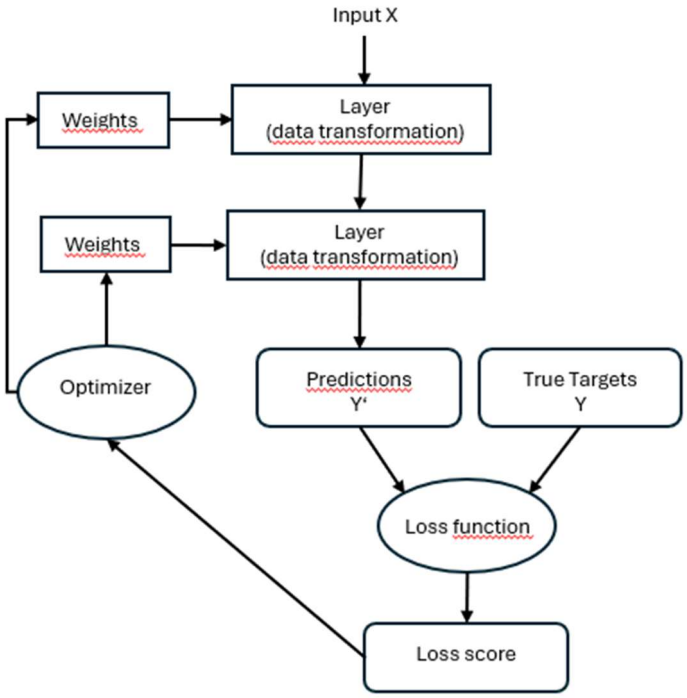
$$\frac{\partial L}{\partial \omega_{11}^{[2]}}.$$

Berechnen Sie diesen Wert im Anschluss. Nutzen Sie hierfür:

1) Den Vektor $y = (0, 1.57)^T$.
2) Falls Sie Teil b) nicht bearbeitet haben: $a^{[2]} = (0, 1.6)^T, a^{[1]} = (0.27, 0)^T$.
3) Für die Aktivierungsfunktion gilt: $\frac{d}{dx}\sigma(x) = \sigma(x)\left(1 - \sigma(x)\right)$.

**Lösungsvorschlag:**

(a) Der prinzipielle Ablauf des Trainings eines FFNN kann wie folgt dargestellt werden:




Nach:
Deep Learning with Python, Second Edition
(Francois Chollet)

Die Trainingsdaten („Input X“) werden durch die Schichten des Netzwerks (die zunächst mit zufälligen Gewichten und Biaswerten initialisiert werden) verarbeitet und resultieren in den Ausgabewerten („Predictions Y‘). Diese werden zusammen mit den wahren Targets („True Targets Y“) in einer Verlustfunktion („Loss function“) aggregiert (je nach Problemstellung z.B. mse (Regression) oder auch binary cross-entropy loss (Klassifikation)). Zur Anpassung der Gewichte wird der Gradient der Verlustfunktion (mit Hilfe von Backpropagation) berechnet und optimiert. Typischerweise werden die Trainingsdaten in Batches zusammengefasst und damit schrittweise eine Optimierung vorgenommen (Mini-Batch Gradient Descent).

Dieses Verfahren wird für mehrere Epochen (wobei eine Epoche ein Durchlauf durch den kompletten Datensatz darstellt) iterativ wiederholt.

(b) Folgende Gewichtsmatrizen können abgelesen werden:

$$W^{[1]} = \begin{pmatrix} -1 & 0 \\ 0.5 & 0 \\ 0 & -1 \end{pmatrix}, W^{[2]} = \begin{pmatrix} 0 & -1 \\ 0 & 2 \end{pmatrix}.$$

Damit ergibt sich:

$$W^{[1]^T} x = \begin{pmatrix} -1 \\ -1 \end{pmatrix}, \sigma \begin{pmatrix} -1 \\ -1 \end{pmatrix} = \begin{pmatrix} 0.27 \\ 0.27 \end{pmatrix} = a^{[1]},$$

$$W^{[2]^T} a^{[1]} = \begin{pmatrix} 0 \\ 0.27 \end{pmatrix}, \sigma \begin{pmatrix} 0 \\ 0.27 \end{pmatrix} = \begin{pmatrix} 0.5 \\ 0.57 \end{pmatrix} = a^{[2]}.$$

(c) Nach der Kettenregel ergibt sich:

$$\frac{\partial L}{\partial \omega_{11}^{[2]}} = \frac{\partial L}{\partial a_1^{[2]}} \times \frac{\partial a_1^{[2]}}{\partial z_1^{[2]}} \times \frac{\partial z_1^{[2]}}{\partial \omega_{11}^{[2]}}.$$

Dabei ist

$$z_1^{[2]} = \omega_{11}^{[2]} a_1^{[1]} + \omega_{21}^{[2]} a_2^{[1]}, a_1^{[2]} = \sigma\left(z_1^{[2]}\right).$$

Daraus ergibt sich:

$$\frac{\partial L}{\partial a_1^{[2]}} = 2\left(a_1^{[2]} - y_1\right),$$

$$\frac{\partial a_1^{[2]}}{\partial z_1^{[2]}} = \sigma'\left(z_1^{[2]}\right) = a_1^{[2]}\left(1 - a_1^{[2]}\right),$$

$$\frac{\partial z_1^{[2]}}{\partial \omega_{11}^{[2]}} = a_1^{[1]}.$$

Und damit:

$$\frac{\partial L}{\partial \omega_{11}^{[2]}} = 2\left(a_1^{[2]} - y_1\right) a_1^{[2]}\left(1 - a_1^{[2]}\right) a_1^{[1]}.$$

Für den Wert $a^{[2]} = (0.5, 0.57)^T$ sowie $a^{[1]} = (0.27, 0.27)^T$ und $y = (0, 1.57)^T$ ergibt sich damit $2 \times (0.5 - 0) \times 0.5 \times (1 - 0.5) \times 0.27 = 0.0675$.

**Aufgabe 8.** *[Informationsverarbeitung in Versicherungen, Datenverarbeitungstechnologien 1] [12 Punkte]*

(a) *[6 Punkte]* Beschreiben Sie zwei typischerweise vorhandene Softwarearchitekturen in Versicherungsunternehmen. Nennen Sie im Anschluss vier generelle Anforderungen an in Versicherungsunternehmen verwendete Software.

(b) *[6 Punkte]* Nennen Sie drei verschiedenen Ebenen, auf denen für ein Softwareprodukt Tests stattfinden. Beschreiben Sie diese im Anschluss.

**Lösungsvorschlag:**

(a) Bei den Software-Architekturen kann zwischen Monolith und Client/Server unterschieden werden:

(i) Monolith: Bei dieser Architektur ist oftmals alles in einem Programm zusammengefasst. Das bezieht sich auf die Fachlichkeit, die Anzeige (GUI) und die Datenspeicherung.

(ii) Bei der Client/Server Architektur ist der Server die zentrale Komponente. Dieser erhält Anfragen von Clients über ein Netzwerk und sendet entsprechende Antworten zurück.

Anforderungen an die zu verwendende Software im Versicherungskontext:

Revisionssicher:

- Grundsätze ordnungsgemäßer Buchführung (BMF GoBD)
- IDW Stellungnahme zur Rechnungslegung: Grundsätze ordnungsgemäßer Buchführung bei Einsatz von Informationstechnologie (IDW-RS-FAIT 1)
- Versicherungsaufsichtliche Anforderungen an die IT (BAFIN VAIT)

Wartbarkeit / Skalierbarkeit:

- Fehler müssen behebbar sein
- Flexibel bei Anwendern und Produkten, d.h. egal ob ein Anwender und ein Produkt oder 1.000 Anwender und 100 Produkte

Flexibel:

- Neue Features einfach implementierbar

Bezahlbar

Andere Nennungen sind möglich.

(b) Für ein Softwareprodukt können Tests auf folgenden Ebenen stattfinden:

(i) Unit Tests: Eine Unit kann ein Modul, eine Funktion oder eine Klasse sein. Der Test ist ein kleines Programm, das prüft, ob der Ausgabewert einer bestimmten Bedingung genügt, z.B. mit einem festen Wert übereinstimmt. Anwendung insbesondere nach erfolgter Implementierung, Update oder Korrektur.

(ii) Integration Tests: Auf dieser Ebene werden zielgerichtet kritische Schnittstellen zwischen Modulen getestet. Die Ausführungszeiten von Integrationstests sind länger im Vergleich mit Unit Tests und auch die Pflege ist aufwändiger.

(iii) Systemtests/Acceptance Tests: Langsam, und mitunter sehr wartungsintensiv. Hilfreich, um die Funktion der Anwendung als Ganzes zu prüfen. Wenn ein Benutzer/Kunde/Unternehmen die Anwendung erhält, werden diese u.a. durchgeführt, um zu überprüfen, ob die Funktionalität den Anforderungen entspricht.

DAV
Deutsche
Aktuarvereinigung e.V.

**Written examination in the subject**

**Actuarial Data Science Basic**

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

on May 23, 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 38 pages and an annex.
- 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, Dr. Jonas Offtermatt, Dr. Felix Spangenberg

**Task 1** *[1.1 Basics, 3.1 Data Mining 1, 4.2 Unsupervised Learning 1] [36 points]*

Your insurance company is planning to expand its product range to meet the different political and environmental preferences of its customers. The aim is to develop sustainable insurance products for environmentally conscious customers and conventional products for customers who place less value on sustainability. Various data points will be used to segment and analyze the customer groups, including demographic data, political preferences, vehicle type, energy consumption, purchasing habits and participation in environmental initiatives.

(a) *[6 points]* Analyze the customer data in the following table:

| | Age | Sex | Income | Political | Vehicle | Energy | Shopping Habits | Environmental |
|---|---|---|---|---|---|---|---|---|
| **1** | 26 | F | 29000.0 | Green | Petrol | Medium | Supermarket | No |
| **2** | 67 | M | 102000.0 | Green | Electric | Low | Organic | Yes |
| **3** | 31 | F | 35000.0 | Center | Diesel | High | Discounter | No |
| **4** | 26 | M | 44000.0 | Conservative | Petrol | Low | Organic | Yes |
| **5** | 57 | M | 72000.0 | Conservative | Diesel | Medium | Discounter | No |
| **6** | 67 | M | 97000.0 | Center | Petrol | Low | Supermarket | No |
| **7** | 24 | F | 18000.0 | Center | Diesel | High | Discounter | No |
| **8** | 74 | M | 106000.0 | Center | Diesel | Low | Discounter | No |

In doing so, go into the various data types and characteristic types.

(b) *[6 points]* What risks do you see with this product? (Name 3 risks)

(c) *[8 points]* Draw a scatterplot that plots the *income* of the customers against their *age* and color-codes their political preference. How many clusters do you expect to find in the data?

(d) *[4 points]* Discuss how the actuarial profession could change as a result of the introduction of such new products. Take into account the challenges arising from the analysis and use of sensitive customer data and the networking of other departments.

(e) *[12 points]* Perform the k-means clustering steps below on the **standardized data below** to segment customers into two groups. Use the attributes age and income:

(i) Select the values $Z_1(-0{,}8 \,/\, -1{,}1), Z_2(1{,}2 / \,1{,}6)$ as the starting centroids.

(ii) Calculate the affiliation of the first **five** data points to the starting centroids. (Use the Euclidean distance as the distance measure. Two points $P_1(x_1 / x_2), P_2(y_1 / y_2)$ then have the distance

$$a = \sqrt[2]{(x_1 - y_1)^2 + (x_2 - y_2)^2}$$

(iii) Update the position of the centroids based on the assigned data points.

*Note:* Use the standardized values from the following table for the calculation:

| Data point | Age (standardized) | Income (standardized) |
|---|---|---|
| 1 | -0,8 | -1,1 |
| 2 | 1,2 | 1,6 |
| 3 | -0,9 | -0,3 |
| 4 | -0,6 | -0,6 |
| 5 | -0,8 | -1 |

**Proposed solution:**

(a) The customer data comprises various data types and characteristic types that are relevant for the segmentation and development of new insurance products:

**Categorical data:** Gender, political preference, vehicle type, energy consumption, purchasing habits, participation in environmental initiatives (Boolean).

**Scalar data:** Age, income.

(b) Risks associated with this product:

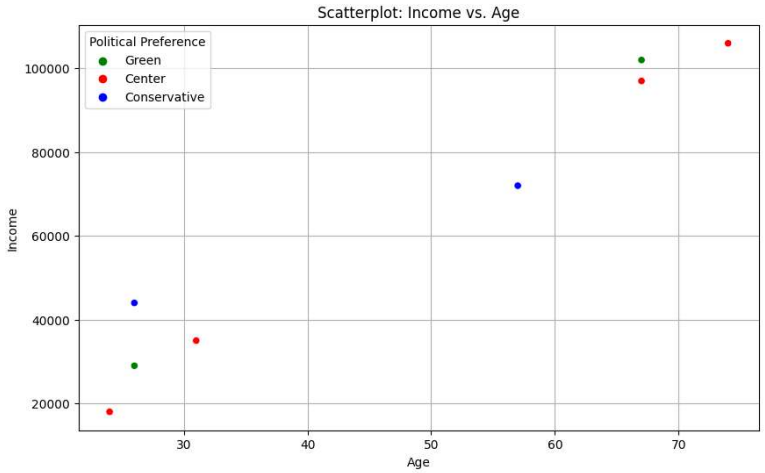
(i) Data protection risks: Increased sensitivity in the processing and storage of environment-related customer data, particularly in relation to the GDPR.

(ii) Market risks: Possible rejection or lower acceptance of the new sustainable products by conservative or price-sensitive customer groups.

(iii) Operational risks: Challenges in integrating new data and production processes that could affect the efficiency and profitability of the new products.

(iv) Reputational risk: Segmentation by political preference can lead to reputational risk, especially if customers may feel that their personal beliefs are not respected or are being used inappropriately. This could lead to a loss of reputation for the company if the segmentation is not communicated transparently and fairly.

(c) ScatterPlot:




Probably 2 clusters.

(d) Change in the actuarial profession:

(i) Actuaries must increasingly carry out data analyses in order to understand customer preferences and develop segmented products.

(ii) Collaboration with other departments such as marketing and IT to integrate data and develop new products.

(iii) Increased responsibility in handling sensitive customer data and compliance with data protection regulations

(iv) Increased demands on the IT knowledge of actuaries to create data analyses using new algorithms and larger data volumes.

(v) ...

(e) Calculation steps

First of all, we calculate the distance from each of the five points to the two starting centroids :$Z_1(-0{,}8\ /\ -1{,}1), Z_2(1{,}2/\ 1{,}6)$

| Data point | Age (standardized) | Income (standardized) | Distance to $Z_1$ | Distance to $Z_2$ | Cluster |
|---|---|---|---|---|---|
| **1** | -0,8 | -1,1 | 0,00 | 3,36 | 1 |
| **2** | 1,2 | 1,6 | 3,36 | 0,00 | 2 |
| **3** | -0,9 | -0,3 | 0,81 | 2,83 | 1 |
| **4** | -0,6 | -0,6 | 0,54 | 2,84 | 1 |
| **5** | -0,8 | -1 | 0,10 | 3,28 | 1 |

Obviously, all three other points belong to the group around the first centroid.

We then calculate the new starting centroids. As there is only one point in group 2, this centroid remains unchanged.

$$Z_2^{(2)} = Z_2(1{,}2\ /\ 1{,}6)$$

Group 1 contains four points. This results in the new centroid:

$$Z_1^{(2)} = \left(\frac{-0{,}8 - 0{,}9 - 0{,}6 - 0{,}8}{4} / \frac{-1{,}1 - 0{,}3 - 0{,}6 - 1}{4}\right) = (-0{,}775/-0{,}75)$$

**Task 2** *[1.1 Basics, 3.1 Data Mining 1, 4.2 Unsupervised Learning 1, 5.2 Data Science Tools] [18 points]*

Your company processes customer data from various sources and must support different data formats. Both technical and data protection requirements must be considered.

(a) *[6 points]* Your company must support various data formats to import and export customer data from other companies. Name two open data formats (other than JSON) that the interface should be able to process and provide a sample data set for each format.

(b) *[6 points]* Customer data must be processed and stored in accordance with the General Data Protection Regulation (GDPR). Explain how storing data in open formats helps your company in accordance with Article 15 GDPR and what this article entails.

(c) *[6 points]* Name and describe 3 tools that can be used both via a user interface and via an R and/or Python interface in order to be able to process and analyze the presumably very large amount of data appropriately. Name the user interface and interfaces in each case.

**Proposed solution:**

(a) Open data formats:

(i) XML (Extensible Markup Language):

Description: XML is a markup language designed for both human and machine readability and is commonly used for data exchange on the web.

Example:

```
<customer>
  <customer number>12345</customer number>
  <name>Max Mustermann</name>
  <age>30</age>
  <insurance type>Car insurance</insurance type
  <premium>450.50</premium>
</Customer>
```

(ii) CSV (CommaSeparated Values):

Description: CSV is a simple format for saving tabular data in which each line is a data record and the individual data fields are separated by commas.

Example:

```
Customer number,name,age,insurance type,premium
12345,Max Mustermann,30,car insurance,450.50
```

(iii) Other examples would be .txt, .html, .odt, .gpx, ... files

(b) Article 15 GDPR:

Content: Article 15 grants data subjects the right to information about the personal data that a company stores about them. This includes information on the purpose of the processing, the categories of data processed and the recipients of the data.

Advantages of open formats:

- Easy data access: Open formats make it easier to provide the requested information, as they are standardized and widely used.
- Interoperability: Data in open formats can be easily exchanged between different systems, which speeds up the fulfillment of requests for information.
- Transparency: By using open formats, the company can ensure that the data provided is understandable for the data subjects.
- This makes it very easy to provide them in a common electronic format.

(c) Possible tools:

**Tableau:**

Description: A powerful data visualization tool that enables complex data analysis.

User interface: Intuitive drag-and-drop interface.

APIs: Tableau-R and tableau-api-lib for Python.

**Jupyter Notebooks:**

Description: An open-source web application that allows you to create and share documents containing live code, equations, visualizations and narrative text.

User interface: Web-based user interface for interactive programming.

APIs: Direct integration with Python (Jupyter) and R (IRkernel).

**RStudio:**

Description: An integrated development environment for R and Python that is used for both statistical analysis and the development of data science applications.

User interface: Comprehensive IDE for R programming.

APIs: Direct integration with R packages and R scripts.

**Power BI:**

Description: Microsoft's business analytics tool that provides detailed insights and visualizations.

User interface: User-friendly drag-and-drop interface for creating interactive reports and dashboards.

APIs: Support for R and Python directly in Power BI desktop through integrated script visualizations.

**KNIME:**

Description: An open-source data analysis, reporting and integration software that simplifies data science workflows.

User interface: Visual workflow-based user interface that makes it possible to create complex data processes without extensive programming.

APIs: Integrated nodes for R and Python that enable scripts to be executed directly within workflows.

Further tool names are possible.

**Task 3** *[1.4 Data Protection 1, 2.3 Information processing in insurance companies, 3.1 Data mining 1] [14 points]*

You are a member of a new project team that will explore and implement the application of machine learning in the insurance company. The team has the task of developing initial concepts that both improve data quality and apply the principles of machine learning in the company's processes. In addition, the aim is to ensure that all regulatory requirements relating to the use of artificial intelligence (AI) in the insurance industry are observed.

(a) *[6 points]* The team has identified that there are some inaccuracies and gaps in the existing customer data. Name three specific measures to improve data quality and explain how you would apply these measures in an insurance company.

(b) *[5 points]* Explain the differences between supervised and unsupervised learning in the context of machine learning methods. Give a concrete example of the use of supervised and unsupervised learning in the insurance industry.

(c) *[3 points]* Name the two basic sources of law that must be observed when using data analysis and artificial intelligence.

**Proposed solution:**

(a) Measures to increase data quality:

**Preventive measures:**

- Avoiding the incorrect recording of data by
  - Use of clear selection lists
  - Automatic verification of entries
  - Allow ignorance / graduated plausibility check
  - Determine the time of data collection
- Introduction of data architecture / company-wide data model
  - Assignment of data to business domains
  - Avoidance of redundancy
  - Use of technical definitions
  - Resolving semantic overlays

**Measures on given data:**

- Measure data quality by comparing it against key figures (find outliers etc.)
- Rule-based on the basis of defined deviation tolerances

(b) **Supervised learning**

In supervised learning, a model is trained using data that has already been labeled or classified (i.e. there are input and target variables). The aim is to make predictions or classifications based on the input data.

Example in the insurance industry: A model for predicting claims in car insurance. Using historical data (e.g. vehicle age, driving behavior, etc.) and the associated claims (target variables), the model can be trained to predict future claims.

**Unsupervised learning**

In unsupervised learning, there are no predefined target variables. The model is trained to recognize patterns and structures in the data.

Example in the insurance industry: Segmentation of customers according to similarities in behavior. An insurance company could, for example, use unsupervised learning to identify customer groups with similar characteristics (e.g. age, risk behavior, premium level), which could then be used for targeted marketing measures.

(c) **1 GDPR** (General Data Protection Regulation)

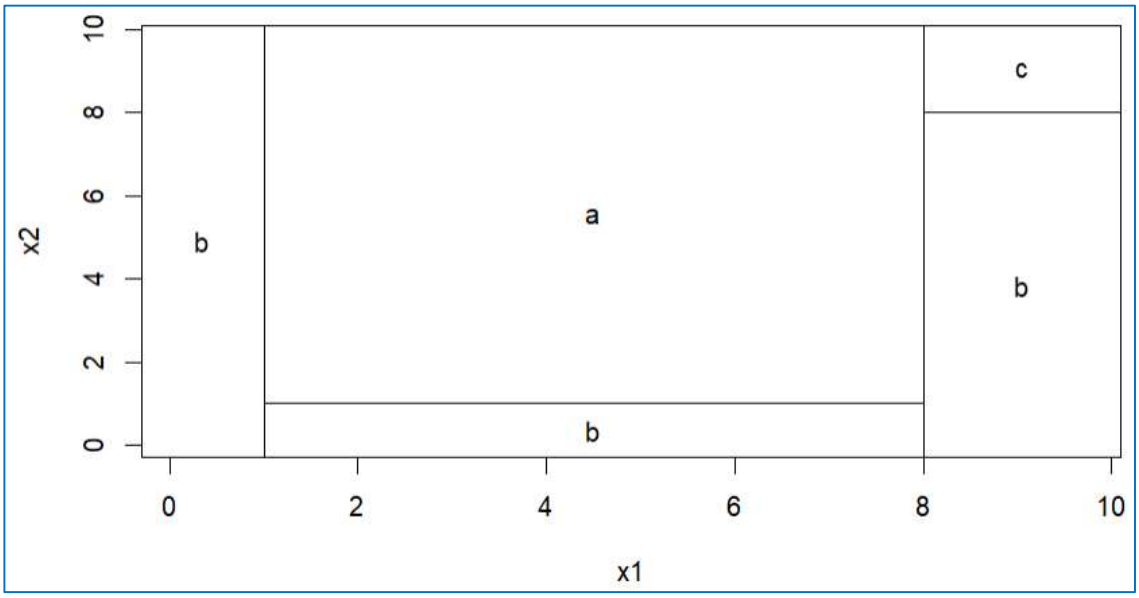
The GDPR regulates the protection of personal data in the European Union. When using AI and data analytics in the insurance industry, it is crucial to ensure that the processing of customer data complies with the requirements of the GDPR, particularly with regard to consent to data use, the right to be forgotten and the protection of sensitive data.

**2. the EU AI Act** (European regulation on the regulation of artificial intelligence)

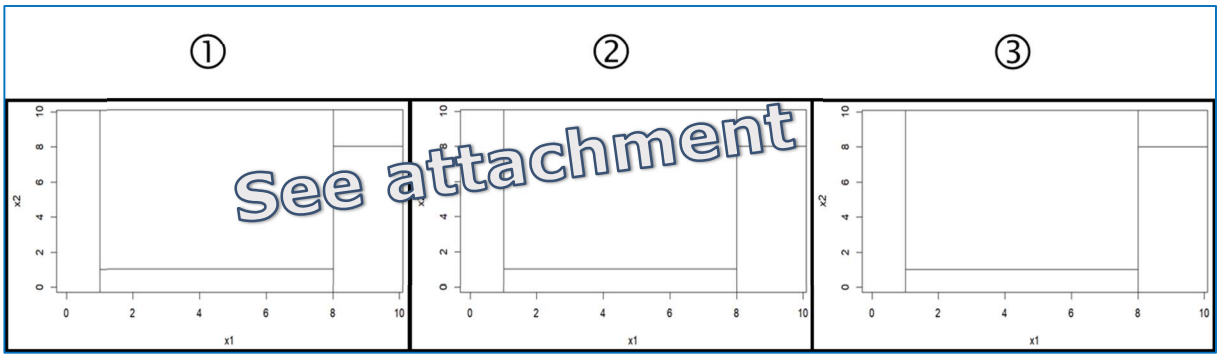
The EU AI Act is a regulatory framework that aims to regulate the development and use of AI systems in the EU. It distinguishes between different risk classes of AI applications, ranging from minimal to unacceptable risk. Insurance companies that use AI must ensure that their systems are transparent, secure and fair and comply with legal requirements.

**Task 4** *[4.1 Supervised learning 1] [27 points]*

(a) *[5 points]* Below you can see a partition of a 2-dimensional feature space created by classification using the CART algorithm. Draw a suitable decision tree. Round numerical values from the diagram below to integer values.




(b) *[6 points]* Note: To answer subtask b), please use the sheet attached to the exam with three graphs ① to ③. Use a ruler for your additions.




Based on the partition graph from subtask a): Draw straight lines that lead back to a partition, with different constraints.

① Draw a straight line so that the new partition could also have been created by the CART algorithm.

② Draw a single straight line so that the new partition does not represent a decision tree.

③ Alternatively, draw two straight lines that result in a substantially different partition that also does not represent a decision tree.

Explain your decisions.

Make sure you also hand in the page you have edited!

(c) *[7 points]* Name 3 advantages and 2 disadvantages of decision trees compared to other ML methods.

Describe in two to three sentences an approach that would result in a decision tree that eliminates one of the disadvantages you mentioned.

(d) *[9 points]* What is the common principle (in one or two sentences) of all so-called ensemble methods?

Explain the idea of bagging or boosting - independent of decision trees - in one sentence each.

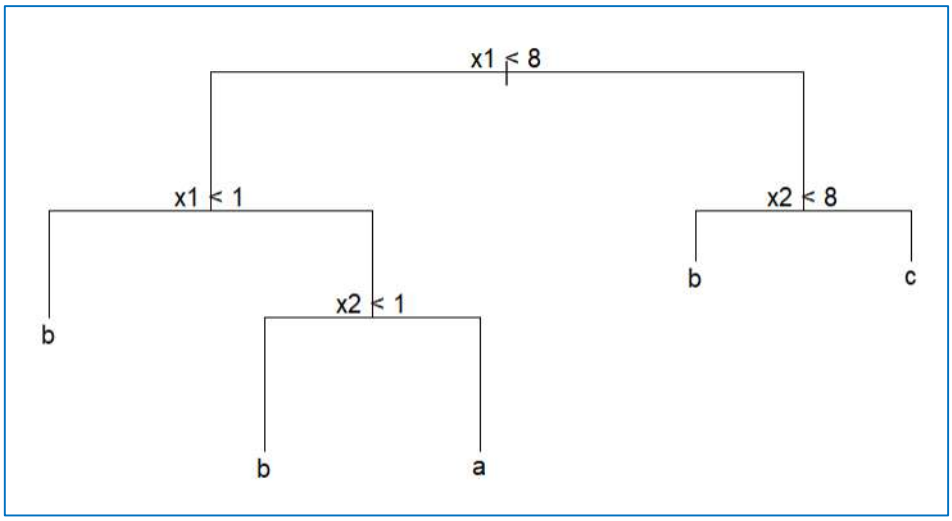
To what extent can bagging or boosting reduce certain disadvantages of decision trees?

How do random forests fit in with these concepts and what is special about random forests and why?

Which essential advantage of decision trees is lost with the random forest?

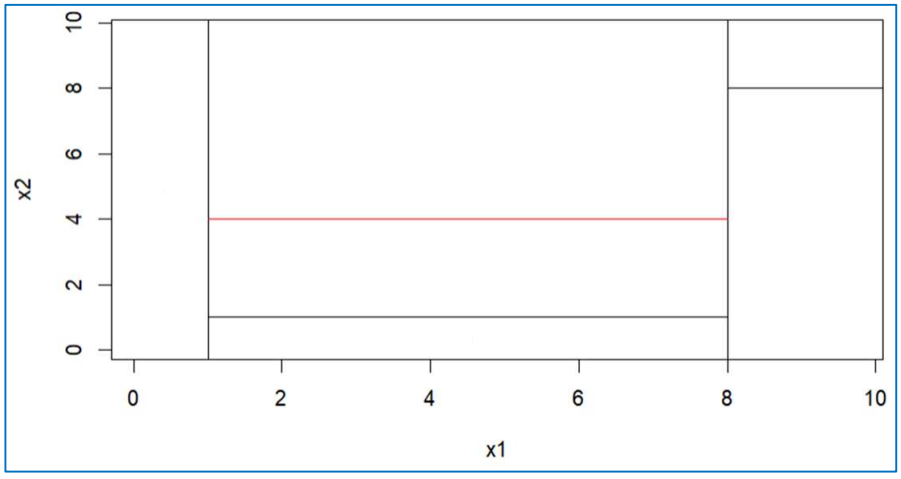
**Proposed solution:**

(a) The following tree shows the check in each case (here with "<"; "≤" would also be possible) and then branches to the left in the case of "yes" and to the right in the case of "no". The respective classification is specified in the leaves according to the partition graphic.

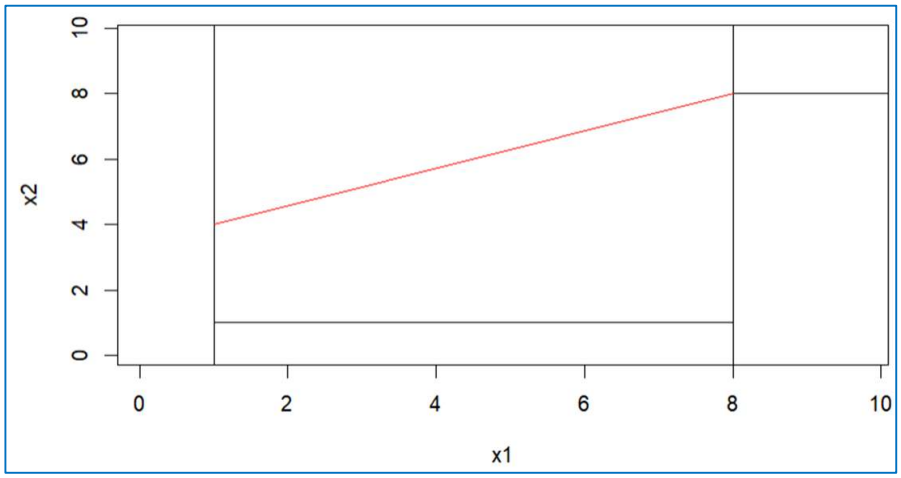



*Note: Other solutions are possible; depending on how the original data (not known here) is distributed, the check "x1<1" can be carried out first (i.e. at the top of the tree) and the check "x1<8" only at the level below.*

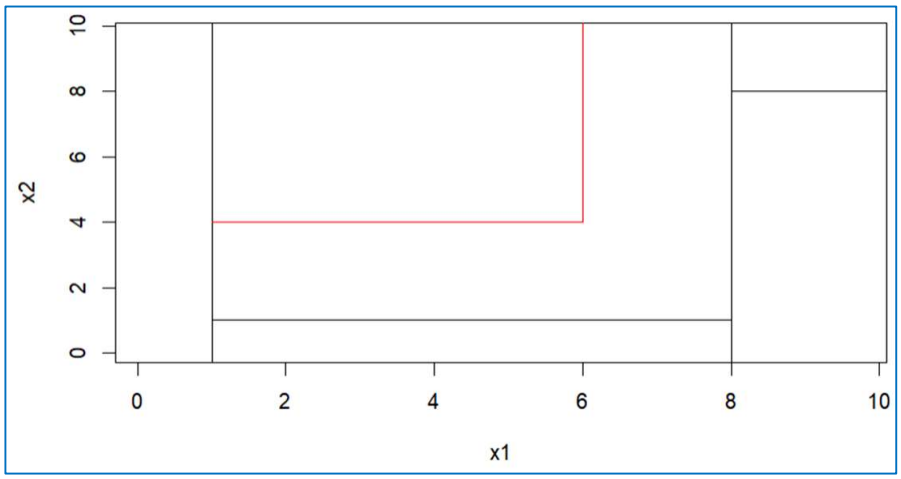
(b) ① An additional axis-parallel line that connects to the existing lines results in a partition that could also have been created by the CART algorithm:




② An additional non-axis-parallel line can never be created by the CART algorithm:




③ An area formed by two additional axis-parallel lines cannot belong to a CART-conditional partition if these two lines do not extend to the edge or to existing lines:




(c) **Advantages of decision trees:** *(it is sufficient to name 3 advantages)*

1. **ease of interpretation:** Decision trees are easy to understand and interpret and therefore easy to communicate. In particular, smaller trees that are not too deep are comparatively easy to visualize. Each node represents a decision that is clearly comprehensible, which makes them particularly attractive for use in areas with strict regulatory requirements.

2. **scaling of the data is superfluous:** Decision trees do not require data normalization or standardization. You can work directly with categorical and numerical data without the need for pre-processing steps such as scaling.

3. **handling missing values:** Decision trees can handle missing data by using surrogate splits for missing attributes or by ignoring missing values during training.

4. **robustness against outliers:** Decision trees are less susceptible to outliers because they divide data into discrete segments and are not distorted by extreme values.

5. **identification of meaningful features:** Decision trees are also good for identifying the most meaningful features.

**Disadvantages of decision trees:** *(it is sufficient to name 2 disadvantages)*

1. **overfitting:** Decision trees tend to adapt strongly to the training data, which can lead to poor generalization to new data.

2. **instability:** Small changes in the training data can lead to significant changes in the structure of the decision tree, which has a negative impact on the interpretability and reliability of the results.

3. **no complex relationships:** Decision trees may have difficulty capturing complex relationships in the data, leading to performance degradation in certain scenarios.

*Note: For full points for the advantages and disadvantages, it is sufficient to name three or two clearly understandable keywords (corresponding to the bolded terms in the list above).*

The problem of overfitting can be mitigated within the area of decision trees by the technique of pruning. This reduces the complexity of a decision tree. Pruning can be carried out in advance (using suitable stop criteria) or retrospectively (by comparing several trees, e.g. via cross-validation).

Note: Although the use of ensemble methods such as random forests also leads to a reduction in overfitting, it leaves the area of decision trees in the narrower sense.

(d) **Ensemble methods:**

Ensemble methods combine the predictions of multiple ML models to improve overall accuracy and provide more robust and powerful predictions than using a single model.

Bagging and boosting are examples of such ensemble methods.

**Bagging** (bootstrap aggregating) combines multiple models trained on different samples of the training data to average their predictions (by averaging or majority voting), thereby reducing the variance and increasing the stability of the model.

**Boosting** is an iterative method in which weak models (models with lower prediction accuracy) are trained one after the other, with each new model correcting the errors of the previous models in order to gradually form a strong overall model.

The aforementioned variance reduction in bagging is accompanied by a reduced tendency to overfitting and thus counteracts this disadvantage of decision trees. Boosting, on the other hand, increases model accuracy through successive error correction, which both reduces bias and improves generalization capability.

A **random forest** is an ensemble of decision trees based on the bagging approach. However, the bootstrapped predictions are usually not independent of each other because dominant features lead to similar splits and thus similar trees. For this reason, a random subset selection is carried out in a second step in the random forest, i.e. a random set of features is selected for each split.

**The disadvantage of random forests** compared to decision trees is the loss of interpretability.

**Task 5** *[5.1 Programming languages for data science] [18 points]*

(a) *[5 points]* In this subtask we use the dataset women from the pre-installed R-package datasets. You will find the following information in the R documentation:

women {datasets} R Documentation

**Average Heights and Weights for American Women**

**Description**

This data set gives the average heights and weights for American women aged 30–39.

**Usage**

```
women
```

**Format**

A data frame with 15 observations on 2 variables.

`[,1]` `height` numeric Height (in)

`[,2]` `weight` numeric Weight (lbs)

Write readable R code with maximum use of `tidyverse` (i.e. in particular `dplyr` and the pipe operator) to accomplish the following task. Carry out all the steps in sequence and do not skip any of them and also follow the notes given at the end:

> Load `tidyverse`; calculate the variable `X1` by calculating the body mass index (BMI) starting from `women` in a new column `bmi`, restricting the table to rows with BMI≥23 and determining the number of these rows (this should be the content of the variable `X1`).

*Notes:*
BMI = "Body weight in kilograms divided by height in meters squared".
Conversion factor inches (in)→ meters: 0.0254.
Conversion factor American pounds (lbs)→ kilograms: 0.45359237.

(b) *[7 points]* What does the output generated by the following R code look like? Use a ruler for the graph.

For graphic properties that are not explicitly defined by the code, please make appropriate reasonable assumptions.

```
c <- tibble(sex=c("m", "m", "w", "w", "w"),
       ins=c(TRUE, FALSE, FALSE, TRUE, FALSE),
       weight=c(75,95,75,90,95),
       height=c(1.8,1.6,1.6,1.75,1.8))

c

c %>% ggplot(aes(x=height, y=weight, shape=factor(sex))) +
      geom_point(size=3) +
      labs(title="What does this plot show?", shape="sex")
```

How does the graphic change if you enter the following command instead?

```
c %>% ggplot(aes(x=height, y=weight, shape=factor(sex), size=3)) +
      geom_point() +
      labs(title="What does this plot show?", shape="sex")
```

Why does this (undesirable) behaviour occur?

(c) *[6 points]* Explain the terms copy-on-modify and modify-in-place using a short R code example. What role do these concepts play in R?

**Proposed solution:**

(a) Code for executing the task:

```
library(tidyverse)

X1 <- women %>%
      mutate(bmi=(0.45359237*weight)/(0.0254*height)^2) %>%
      filter(bmi>=23) %>%
      nrow
```

Note: The information from the R documentation cited in the task can be found in this form at https://search.r-project.org/R/refmans/datasets/html/women.html.

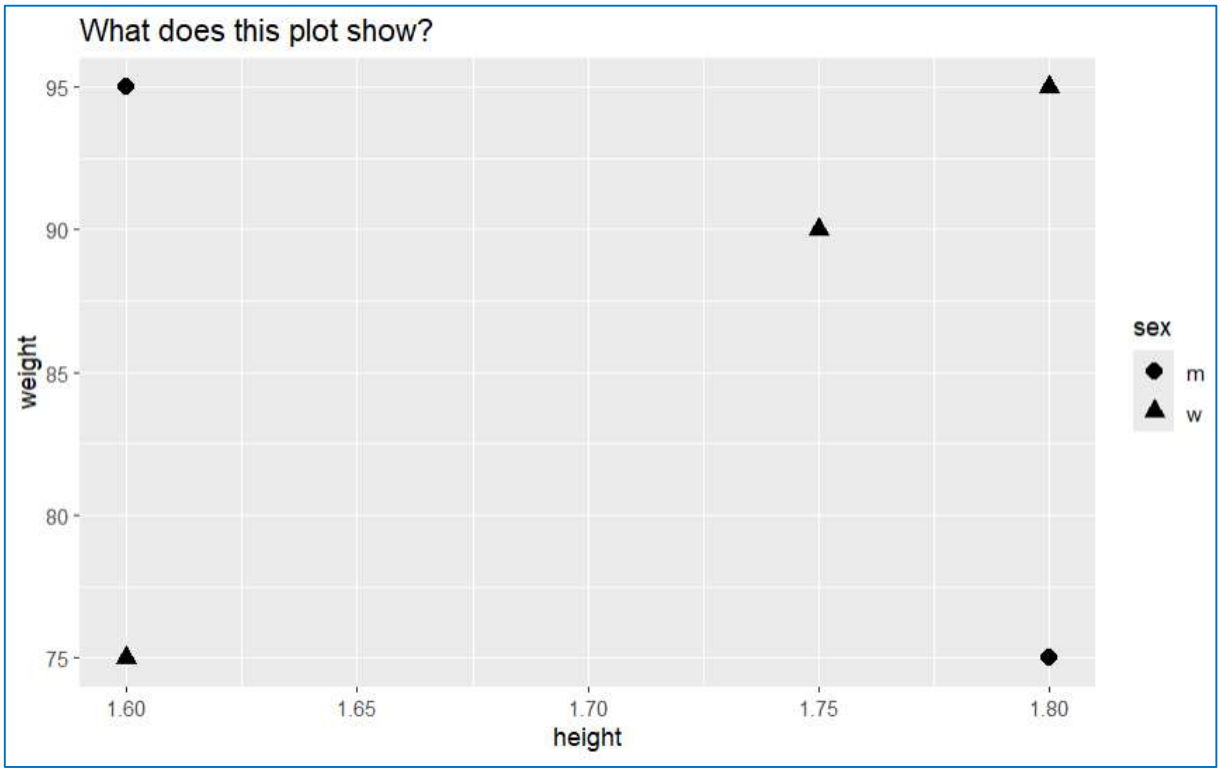
(b) The call of c returns the following **tabular representation:**

A tibble: 5 × 4

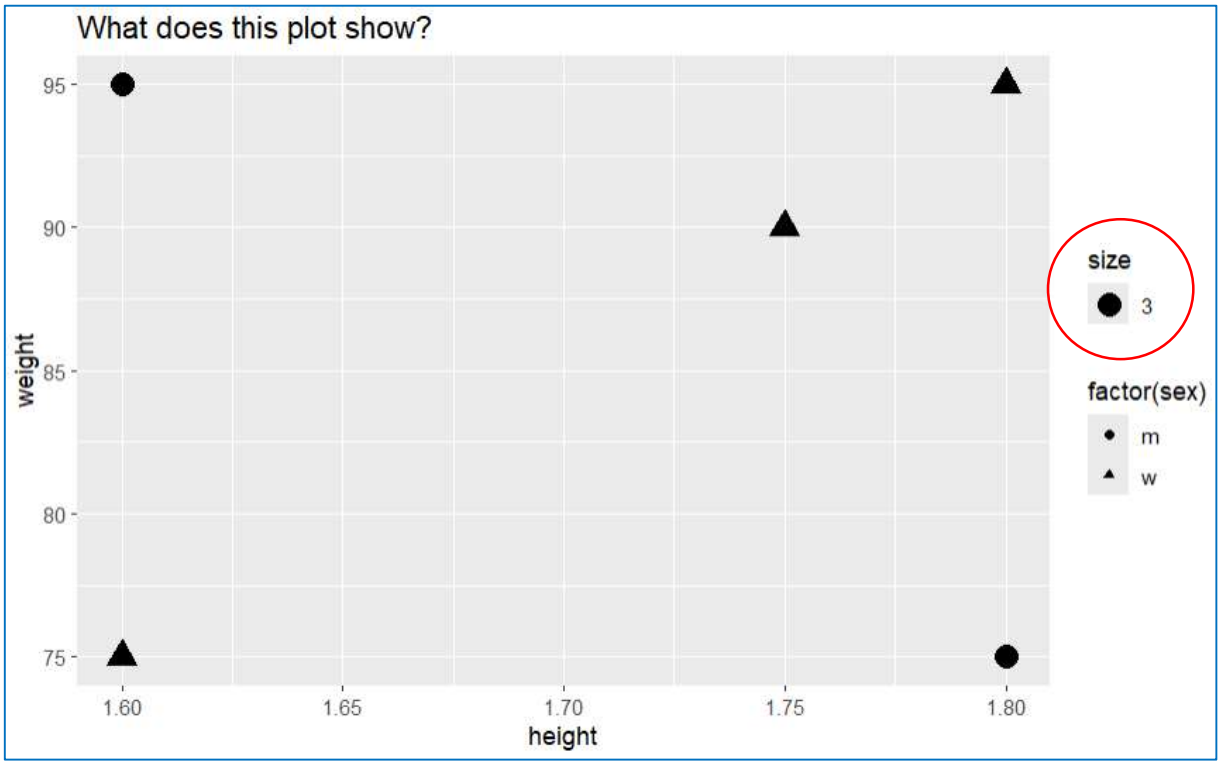
| sex <chr> | ins <lgl> | weight <dbl> | height <dbl> |
|---|---|---|---|
| m | TRUE | 75 | 1.80 |
| m | FALSE | 95 | 1.60 |
| w | FALSE | 75 | 1.60 |
| w | TRUE | 90 | 1.75 |
| w | FALSE | 95 | 1.80 |

5 rows

The **graphic output** looks as follows:




The **alternative `ggplot` call**, on the other hand, returns an additional legend for a variable `size` (highlighted by a red circle in the following example graphic). This is because in this case the assignment "`size=3`" is incorrectly included within the `aes` function, i.e. it is interpreted as an aesthetic mapping for a constant "variable" 3, which is to be displayed via the `size` of the data points or lines; the size of the "points" is not influenced by this value 3 (e.g. the same graphic is obtained with the value 10 - apart from the 10 in the legend). In the correct `ggplot` call, however, `size` is understood as an argument for `geom_point()` and influences the size of the "points".




(c) Modify-in-place and copy-on-modify are concepts that are used in R to manage the allocated memory space when modifying objects. The two methods differ in terms of the economy of memory requirements.

- With **modify-in-place**, an object that already exists in R is changed by modifying its assigned value in memory. No copy is created and the old status is no longer available. Depending on the memory requirements of the newly bound value, more memory may be used or released again.

- With **copy-on-modify**, a copy of the original object is created in the memory by modifying the value of an object, i.e. the previous object continues to exist in its previous form in the memory. In principle, this requires more memory.

Modify-in-place is generally used for objects for which only one unique binding exists (i.e. which are only referenced by one name). In the following example, the value 3 no longer exists in the memory after the program has been executed, but the binding x continues to reference the original object with the modified value 4.

```
x <- c(1, 2, 3)
x[3] <- 4
x

## [1] 1 2 4
```

Modify-in-place is also always used for environments (objects of type `environment`).

Otherwise, R always uses copy-on-modify. For example, after an atomic vector receives a second binding, both names initially reference the same values in the memory. If a value of the object is now modified by subassignment to one of the two names, both the original and the modified object exist with the respective name as a reference in the memory (even if many elements of the two vectors still have the same values). In the following example, both an object with values `c(1, 2, 3)` and an object with values `c(1, 2, 4)` are in the memory after the program has been executed.

```
x <- c(1, 2, 3)
y <- x
y[[3]] <- 4
x

## [1] 1 2 3
```

**Task 6** *[Data Management 1, Analytics 1] [25 points]*

(a) *[2 points]* You receive the necessary data in JSON files for the evaluation of two competing models. However, importing the files fails. Here is the content of one of the JSON files:

```
{
  "model_id": "djkzqod6_2024",
  "model_columns: ["id i", "score s", "prob p", "target t"],
  "model_eval": [
     {"i": 1, "s": 150, "p": 0.05, "a": 0},
     {"i": 2, "s": 250, "p": 0.1, "a": 0},
     {"i": 3, "s": 370, "p": 0.3, "a": 1},
     {"i": 4, "s": 550, "p": 0.7, "a": 0},
     {"i": 5, "s": 650, "p": 0.8, "a": 0},
     {"i": 6, "s": 750, "p": 0.9, "a": 1, "final": True}],
     "validated": 2024-09-27
}
```

Identify errors in this content (according to the JSON standard ECMA-404 or RCF 8259).

(b) *[4 points]* After the necessary corrections, the data should be saved on a hard disk. What should generally be taken into account? The information should be saved as text files. Amongst other things, please refer in particular to the ASCII, ANSI and UTF-8 formats.

(c) *[4 points]* Define the following terms for the use case of binary classification

Confusion matrix, precision, sensitivity and specificity.

(d) *[10 points]* Extract the following data records from the JSON from task a) after making the necessary corrections (a target of 1 means a failure according to the model under consideration):

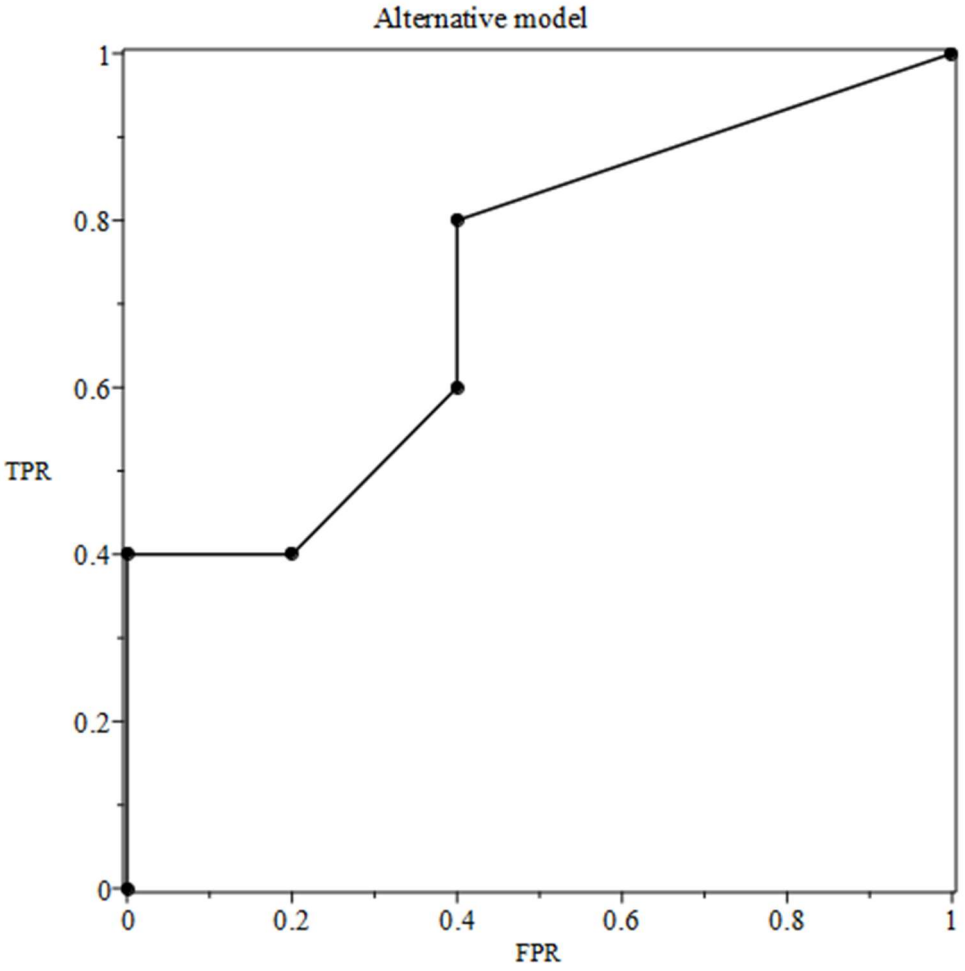
| **ID** | **Score (s)** | **Probability (p)** | **Target (a)** |
|---|---|---|---|
| 1 | 150 | 0.05 | 0 |
| 2 | 250 | 0.1 | 0 |
| 3 | 370 | 0.3 | 1 |
| 4 | 550 | 0.7 | 0 |
| 5 | 650 | 0.8 | 0 |
| 6 | 750 | 0.9 | 1 |

Construct a corresponding ROC chart with an associated ROC curve. Then calculate the AUC value.

*Note:* The calculation steps must be traceable. The following should apply for a value $p$ and a threshold value $\alpha > 0$ :

$$Klassifizierung = \begin{cases} 0\,, falls\ p \leq\ \alpha \\ 1, falls\ p >\ \alpha \end{cases}$$

(e) [5 points] Your colleagues have already created a corresponding ROC diagram for an alternative model:




Calculate the AUC value for this model. Then compare both models with regard to the AUC score. Which model would you prefer? Is it sufficient to consider the AUC value alone? Name two other methods of analysis and describe them.

*Note:* If part d) has not been processed, an AUC value of 0.8 can be assumed for this part. This value does not represent the correct result of part d)!

**Proposed solution:**

(a) According to the specified standards, the following errors are found in the JSON (where "{" is in line 1 of the above specification):

a. Missing apostrophe for the string "model_columns" in line 3

b. Missing dot in line 7 for the number 0.3

c. In line 10, the keyword is true, not True

d. A date type is not provided for in JSON. The date in line 11 must therefore be specified as a string.

Note: The customized JSON can be validated e.g. via jsonlint.com.

(b) When storing on a digital medium (e.g. hard disk), the coding used must be taken into account, especially for text-based files. Historically, ASCII should be mentioned here as an example. Here the character encoding consists of a total of 128 characters. ANSI is an extension (from 7 to 8 bits), which means that a total of 256 different characters are possible. The UTF-8 standard is the most widely used encoding for Unicode characters today. A different number of bytes is used to store characters. The specification of an encoding is important, as correct processing can only take place if the correct encoding format is specified.

(c) Definition of terms:

a. Confusion matrix: In the binary classification case, this matrix compares the key figures TP (True Positive), FP (False Positive), FN (False Negative), TN (True Negative). For example, TP means that an observation in the data set with a positive (=1) feature was also evaluated as positive (=1) by the model.

As an example, a confusion matrix can be given in the following form:

| | | Predicted (predicted) | |
|---|---|---|---|
| | | Yes | No |
| Actual (actual) | Yes | TP | FN |
| | No | FP | TN |

b. Precision:

$$Präzision = \frac{TP}{TP + FP}$$

c. Sensitivity (sensitivity):

$$Sensitivität\ (Recall) = \frac{TP}{TP + FN}$$

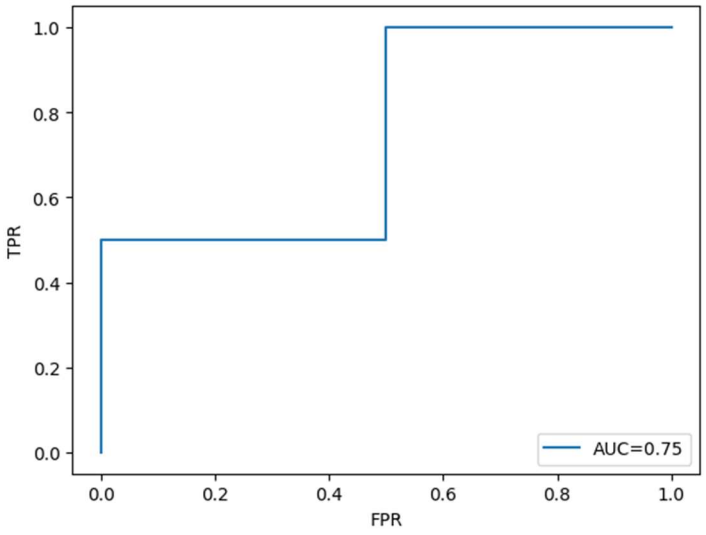
d. Specificity:

$$Spezifität = \frac{TN}{TN + FP}$$

(d) The table shown is already sorted in ascending order according to the "Probability" column. The key figures required for the ROC curve are the True Positive Rate (TPR), which is the same as the sensitivity. The False Positive Rate (FPT) is defined by 1 - Specificity. The ROC curve can be constructed using the following threshold values (example):

| ID | Threshold value | TP | FN | FP | TN | TPR | FPR |
|---|---|---|---|---|---|---|---|
| 1 | 1 | 0 | 2 | 0 | 4 | 0 | 0 |
| 2 | 0.89 | 1 | 1 | 0 | 4 | 0.5 | 0 |
| 3 | 0.79 | 1 | 1 | 1 | 3 | 0.5 | 0.25 |
| 4 | 0.69 | 1 | 1 | 2 | 2 | 0.5 | 0.5 |
| 5 | 0.29 | 2 | 0 | 2 | 2 | 1 | 0.5 |
| 6 | 0.09 | 2 | 0 | 3 | 1 | 1 | 0.75 |
| 7 | 0 | 2 | 0 | 3 | 0 | 1 | 1 |

This results in the following ROC curve:




The AUC value can be determined geometrically from this:

$$AUC = 0.5 \times 0.5 + 0.5 \times 1 = 0.75$$

(e) For the alternative model, the AUC value can be determined geometrically:

$$AUC = 2 \times 0.2 \times 0.4 + 0.6 \times 0.8 + 0.5 \times 0.2 \times 0.2 + 0.5 \times 0.6 \times 0.2 = 0.72$$

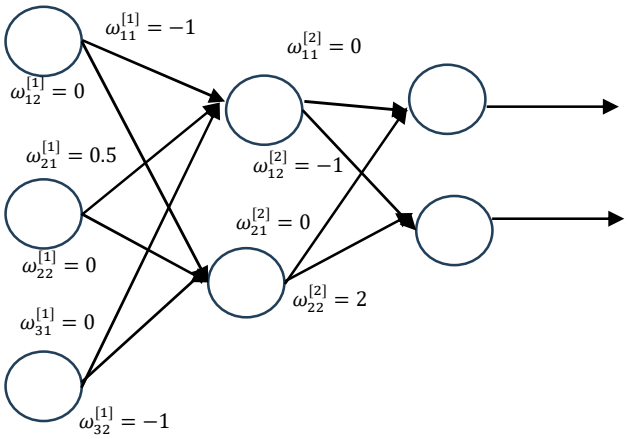
Since the value of 0.72 for the alternative model is lower than the value of 0.75 for the model from part d), the model from part d) according to AUC should be preferred. Other criteria can also be examined:

- Calibration: How well do the calculated probabilities match the observed frequencies? Calibration plots, for example, can be used for this purpose.
- Lift plots (and key figures based on them, e.g. maximum lift point) to better understand the behavior of the models over the data set and to examine critical areas.

**Task 7** *[Deep Learning 1] [30 points]*

(a) *[7 points]* Explain the basic procedure for training a Feed Forward Neural Network (FFNN). Create a sketch and describe important elements.

(b) *[8 points]* Given the following neural network:




Where $\omega_{i,j}^{[k]}$ stands for a weight from node $i$ in layer $k$ to node $j$ in layer $.k+1$

The sigmoid function should be used as the activation function for all layers, i.e.

$$\sigma(x) = \frac{1}{1 + e^{-x}}.$$

Specify the required weight matrices and perform a forward step with the input vector $x = (1, 0, 1)^T$ .

(c) *[15 points]* For a backpropagation, the loss function

$$L = \left\| a^{[2]} - y \right\|_2^2 = \left( a_1^{[2]} - y_1 \right)^2 + \left( a_2^{[2]} - y_2 \right)^2$$

be taken into account. This refers to

$$a^{[2]} = \left( a_1^{[2]}, a_2^{[2]} \right)^T$$

the output of the output layer and

$$y = (y_1, y_2)^T$$

the target value (the expected output) of the network.

The gradient of the loss function L is of central importance for updating the weights. To do this, derive a formula for the partial derivative of the loss function L according to the weight$\omega_{11}^{[2]}$ (see sketch in b) ), i.e.

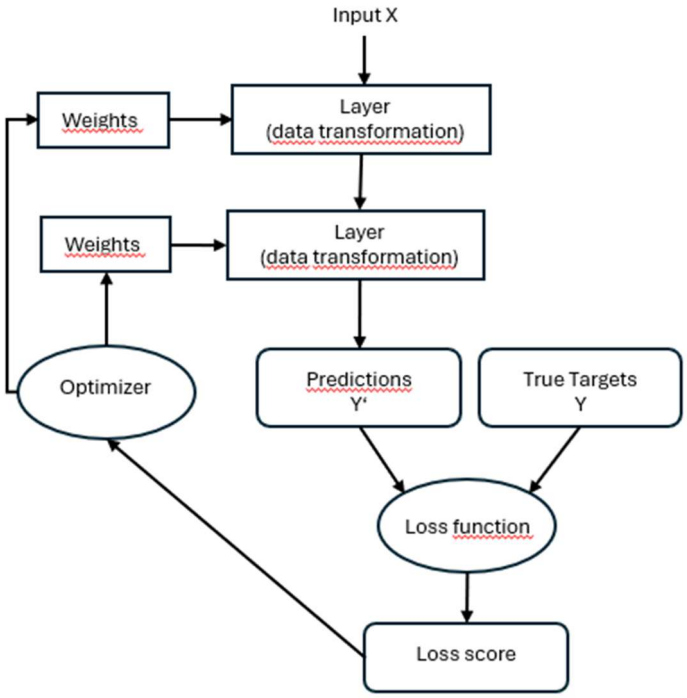
$$\frac{\partial L}{\partial \omega_{11}^{[2]}}.$$

Calculate this value afterwards. Use for this:

1) The vector $y = (0, 1.57)^T$.
2) If you have not completed part b): $a^{[2]} = (0, 1.6)^T, a^{[1]} = (0.27, 0)^T$.
3) The following applies to the activation function: $\frac{d}{dx}\sigma(x) = \sigma(x)\left(1 - \sigma(x)\right)$.

**Proposed solution:**

(a) The basic process of training an FFNN can be described as follows:




Nach:
Deep Learning with Python, Second Edition
(Francois Chollet)

The training data ("Input X") is processed by the layers of the network (which are initially initialized with random weights and bias values) and results in the output values ("Predictions Y"). These are aggregated together with the true targets ("True Targets Y") in a loss function (depending on the problem, e.g. mse (regression) or binary cross-entropy loss (classification)). To adjust the weights, the gradient of the loss function is calculated (using backpropagation) and optimized. Typically, the training data is combined in batches and thus optimized step by step (mini-batch gradient descent).

This procedure is repeated iteratively for several epochs (whereby one epoch represents a run through the complete data set).

(b) The following weight matrices can be read off:

$$W^{[1]} = \begin{pmatrix} -1 & 0 \\ 0.5 & 0 \\ 0 & -1 \end{pmatrix}, W^{[2]} = \begin{pmatrix} 0 & -1 \\ 0 & 2 \end{pmatrix}.$$

This results in:

$$W^{[1]^T} x = \begin{pmatrix} -1 \\ -1 \end{pmatrix}, \sigma \begin{pmatrix} -1 \\ -1 \end{pmatrix} = \begin{pmatrix} 0.27 \\ 0.27 \end{pmatrix} = a^{[1]},$$

$$W^{[2]^T} a^{[1]} = \begin{pmatrix} 0 \\ 0.27 \end{pmatrix}, \sigma \begin{pmatrix} 0 \\ 0.27 \end{pmatrix} = \begin{pmatrix} 0.5 \\ 0.57 \end{pmatrix} = a^{[2]}.$$

(c) According to the chain rule:

$$\frac{\partial L}{\partial \omega_{11}^{[2]}} = \frac{\partial L}{\partial a_1^{[2]}} \times \frac{\partial a_1^{[2]}}{\partial z_1^{[2]}} \times \frac{\partial z_1^{[2]}}{\partial \omega_{11}^{[2]}}.$$

This is

$$z_1^{[2]} = \omega_{11}^{[2]} a_1^{[1]} + \omega_{21}^{[2]} a_2^{[1]}, a_1^{[2]} = \sigma\left(z_1^{[2]}\right).$$

This results in:

$$\frac{\partial L}{\partial a_1^{[2]}} = 2\left(a_1^{[2]} - y_1\right),$$

$$\frac{\partial a_1^{[2]}}{\partial z_1^{[2]}} = \sigma'\left(z_1^{[2]}\right) = a_1^{[2]}\left(1 - a_1^{[2]}\right),$$

$$\frac{\partial z_1^{[2]}}{\partial \omega_{11}^{[2]}} = a_1^{[1]}.$$

And thus:

$$\frac{\partial L}{\partial \omega_{11}^{[2]}} = 2\left(a_1^{[2]} - y_1\right) a_1^{[2]}\left(1 - a_1^{[2]}\right) a_1^{[1]}.$$

This results in $2 \times (0.5 - 0) \times 0.5 \times (1 - 0.5) \times 0.27 = 0.0675$ for the value $a^{[2]} = (0.5, 0.57)^T$ as well as $a^{[1]} = (0.27, 0.27)^T$ and $y = (0, 1.57)^T$.

**Task 8** *[Information processing in insurance, data processing technologies 1] [12 points]*

(a) *[6 points]* Describe two typically existing software architectures in insurance companies. Then name four general requirements for software used in insurance companies.

(b) *[6 points]* Name three different levels at which tests are carried out for a software product. Then describe them.

**Proposed solution:**

(a) A distinction can be made between monolith and client/server software architectures:

(i) Monolith: With this architecture, everything is often combined in one program. This refers to the functionality, the display (GUI) and the data storage.

(ii) In the client/server architecture, the server is the central component. This receives requests from clients via a network and sends corresponding responses back.

Requirements for the software to be used in the insurance context:

Audit-proof:

- Principles of proper accounting (BMF GoBD)
- IDW Statement on Accounting: Generally Accepted Accounting Principles for the Use of Information Technology (IDW-RS-FAIT 1)
- Insurance supervisory requirements for IT (BAFIN VAIT)

Maintainability / scalability:

- Errors must be correctable
- Flexible in terms of users and products, i.e. whether one user and one product or 1,000 users and 100 products

Flexible:

- New features are easy to implement

Affordable

Other nominations are possible.

(b) For a software product, tests can take place at the following levels:

(i) Unit tests: A unit can be a module, a function or a class. The test is a small program that checks whether the output value satisfies a certain condition, e.g. matches a fixed value. Used in particular after implementation, update or correction.

(ii) Integration tests: At this level, critical interfaces between modules are tested in a targeted manner. The execution times of integration tests are longer compared to unit tests and maintenance is also more complex.

(iii) System tests/acceptance tests: Slow and sometimes very maintenance-intensive. Helpful to check the function of the application as a whole. When a user/customer/company receives the application, these are carried out, among other things, to check whether the functionality meets the requirements.