DAV

DEUTSCHE
AKTUARVEREINIGUNG e.V.

Schriftliche Prüfung im Spezialwissen

Actuarial Data Science Basic

gemäß Prüfungsordnung 5
der Deutschen Aktuarvereinigung e. V.

am 24.05.2024

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 33 Seiten.
- Alle Antworten sind zu begründen und bei Rechenaufgaben muss der Lösungsweg ersichtlich sein.

Mitglieder der Prüfungskommission:

Axel Kiermaier, Dr. René Külheim, Dr. Jonas Offtermatt, Dr. Felix Spangenberg

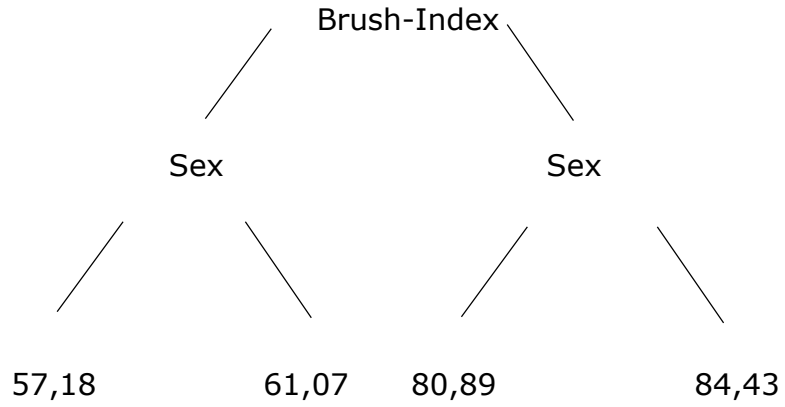
Aufgabe 1 [Datenmanagement 1, Informationsverarbeitung in Versicherungen, Überwachtes Lernen 1] (27 Punkte)

Gegeben seien folgende Schadensdaten für das Produkt Pay-as-you-brush aus Ihrem Unternehmen. Bei diesem Produkt handelt es sich um eine Zahnzusatzversicherung, wobei der Preis gemäß einem Brush-Index angepasst wird. Der Brush-Index wird über die Rückmeldung einer smarten Zahnbürste ermittelt.

Customer_ID	Policy_Number	Brush_Index	Sex	Age	Premium	Claims
CI1	11.84579	Low	Female	22	88.44	84.43
CI2	11.95096	Low	Male	47	95.70	80.89
CI3	11.28109	High	Other	64	115.81	61.07
CI4	11.77030	High	Female	42	78.10	57.11
CI5	11.50399	High	Other	55	107.50	43.05
CI6	11.26674	Low	Other	57	87.33	38.76
CI7	11.39113	Low	Other	54	80.85	61.66
CI8	11.32006	Low	Female	58	118.68	81.94
CI9	11.41755	Low	Female	51	108.54	72.54
CI10	11.42011	Medium	Male	44	88.37	57.18

a) [4 Punkte] Um die Daten maschinell verarbeiten zu können, sollen diese in einem maschinenlesbaren Format abgespeichert werden. Schreiben Sie die ersten drei Zeilen der obigen Tabelle im csv-Format auf.

b) [4 Punkte] Die Höhe der Schäden soll mittels eines Baumverfahrens abgeschätzt werden. In einem ersten Test wurden lediglich die Merkmale Brush_Index und Sex, sowie die Datensätze mit der Customer-ID CI1, CI2, CI3 und CI10 verwendet, um einen unvollständigen Entscheidungsbaum zu erzeugen. Skizze des erzeugten Baumes:



Übertragen Sie die Skizze des Baums auf Ihre Lösungsblätter und ergänzen Sie auf den Linien die entsprechenden Merkmalskriterien.

c) [9 Punkte] Gehen Sie davon aus, dass die obigen Daten in einem R-Dataset mit Namen insurance_data vorliegen. Gegeben sei folgender R-Code:

```
# Create data for the bar plot
plot_data <- insurance_data %>%
  group_by(Sex, Brush_Index) %>%
  summarize(Mean_Claims = mean(Claims))

# Plot the bar plot
ggplot(plot_data, aes(x = Sex, y = Mean_Claims, fill = Brush_Index)) +
  geom_col(position = "dodge") +
  labs(title = "Claims by Sex and Brush Index", y = "Mean Claims")
```

Zeichnen Sie das Schaubild, welches von diesem Code erzeugt wird.

Hinweise: Der Befehl print(`plot_data`) erzeugt:

```
# A tibble: 6 × 3
# Groups:   Sex [3]
  Sex    Brush_Index Mean_Claims
  <chr>  <fct>             <dbl>
1 Female Low                79.6
2 Female High               57.1
3 Male   Low                80.9
4 Male   Medium             57.2
5 Other  Low                50.2
6 Other  High               52.1
```

`Position = “dodge”` führt dazu, dass Balken in einem Balkendiagramm nebeneinander gedruckt werden.

d) [10 Punkte] Ihre Kollegen wollen die Datenaufbereitung in Excel durchführen und die Excel-Datei im lokalen File-System ablegen. Sie plädieren für ein R-Notebook und dafür, dieses in einem zentralen Code-Repository abzuspeichern. Erläutern Sie, warum dieses Vorgehen die generellen Anforderungen an Software besser erfüllt. Gehen Sie dabei auf die Kriterien Revisionssicherheit, Wartbarkeit, Skalierbarkeit, Flexibilität und Bezahlbarkeit ein.

Lösungsvorschlag:

a) Es gibt mehrere Möglichkeiten, die Daten in einer csv-Datei zu speichern. Für die volle Punktzahl ist es nötig, auch die Zeilennamen in der csv zu speichern.

Gültig wäre also:

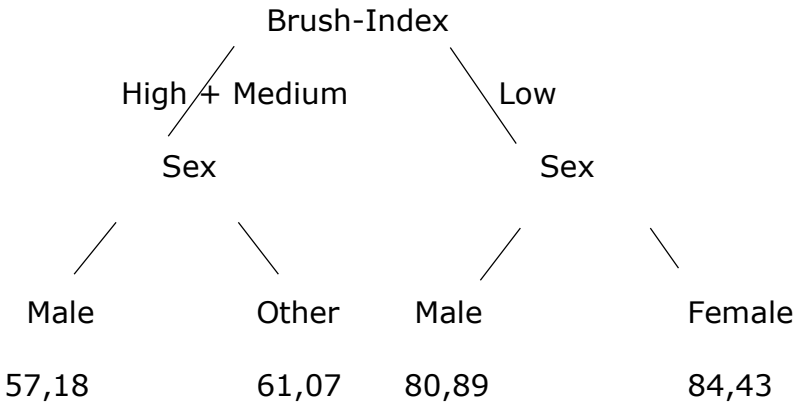
```
"","Customer_ID","Policy_Number","Brush_Index","Sex","Age","Premium",
"Claims"
"1","CI1","11.84579","Low","Female",22,88.44,84.43
"2","CI2","11.95096","Low","Male",47,95.7,80.89
```

Oder auch:

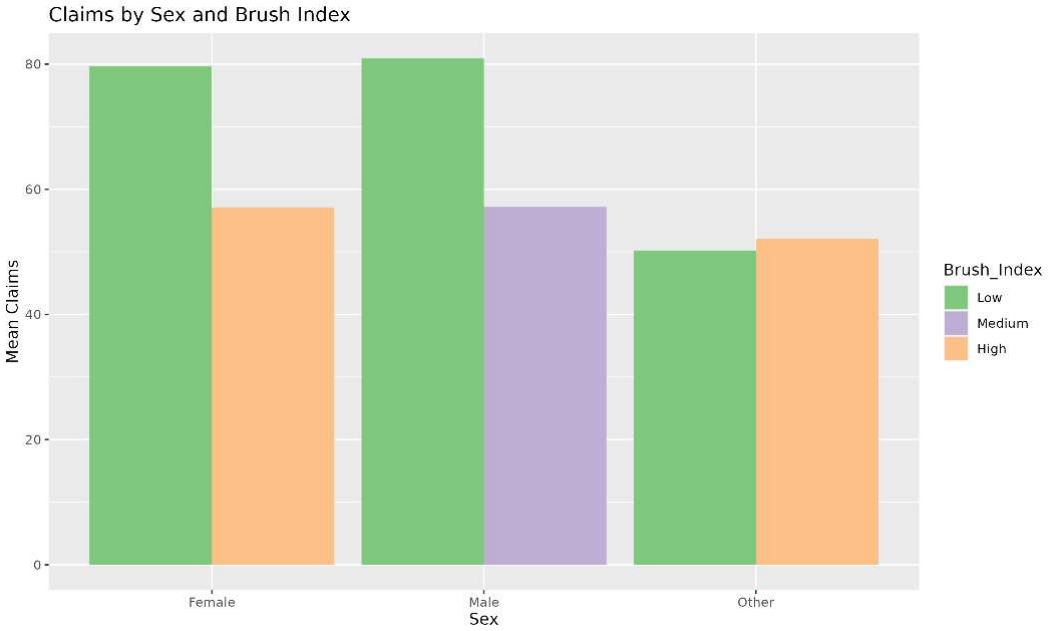
```
"";"Customer_ID";"Policy_Number";"Brush_Index";"Sex";"Age";"Premium";
"Claims"
"1";"CI1";"11.84579";"Low";"Female";22;88,44;84,43
"2";"CI2";"11.95096";"Low";"Male";47;95,7;80,89
```

Es ist auch zulässig, die erste Spalte mit der Nummerierung wegzulassen.

b)



c) Der Code erzeugt folgendes Schaubild:



d) Hier sind einige Gründe, warum die Verwendung eines R-Notebooks in einem zentralen Code-Repository im Vergleich zur Verwendung von Excel-Dateien die generellen Anforderungen an Software besser erfüllt:

Revisionssicherheit:

R-Notebooks sind textbasiert und können problemlos in Versionskontrollsystemen wie Git gespeichert und verwaltet werden. Dies ermöglicht eine genaue Nachverfolgung von Änderungen, die Zusammenarbeit im Team und die Wiederherstellung früherer Versionen.

Wartbarkeit:

In R-Notebooks können Sie den Code dokumentieren und kommentieren, was die Lesbarkeit und Wartbarkeit des Codes verbessert. Dies erleichtert auch anderen Teammitgliedern die Zusammenarbeit und das Verständnis des Codes.

Skalierbarkeit:

R ermöglicht die Automatisierung von Aufgaben und die Verarbeitung großer Datenmengen. Im Gegensatz zu Excel sind Sie in R nicht auf manuelle Schritte und begrenzte Datengrößen beschränkt.

Flexibilität:

Mit R-Notebooks können Sie verschiedene R-Pakete und externe Datenquellen integrieren, um anspruchsvolle Analysen und Visualisierungen durchzuführen. Die Flexibilität von R ermöglicht es Ihnen, Ihre Analyse an spezifische Anforderungen anzupassen.

Bezahlbarkeit:

R ist eine Open-Source-Plattform und kostenlos verfügbar, im Gegensatz zu Excel, für das oft kostenpflichtige Lizenzen erforderlich sind. Dies senkt die Kosten für die Software und ermöglicht den Zugang zu leistungsstarken Analysetools ohne zusätzliche Kosten.

Insgesamt bieten R-Notebooks eine bessere Struktur und Organisation für datenbasierte Aufgaben und erleichtern die Einhaltung bewährter Softwareentwicklungsmethoden. Dies ist besonders wichtig, wenn es darum geht, Datenanalysen in Unternehmen oder Forschungsumgebungen durchzuführen, bei denen Transparenz, Nachverfolgbarkeit und Zusammenarbeit von großer Bedeutung sind.

Aufgabe 2 [Grundlagen, Datenschutz 1, Datenmanagement 1, Informationsverarbeitung in Versicherungen] (18 Punkte)

Ihr Unternehmen möchte ein bestehendes Versicherungsprodukt neu auflegen. Erstmalig sollen hierfür Machine-Learning-Verfahren verwendet werden, um die Prämien genauer kalkulieren zu können.

a) [9 Punkte] Um die Prämien berechnen zu können, sollen die schon vorhandenen Daten um unstrukturierte Daten erweitert werden. Erläutern Sie den Unterschied zwischen strukturierten, unstrukturierten und semistrukturierten Daten und nennen Sie für jeden Datentyp ein Beispiel.

b) [3 Punkte] Nennen Sie drei Geschäftsprozesse von Versicherungsunternehmen, aus denen Sie für das oben genannte Produkt noch weitere Daten für die Kalkulation der Prämien gewinnen könnten.

c) [6 Punkte] Nennen Sie die zentrale Rechtsquelle für Datenschutz innerhalb Europas und erläutern Sie, wieso es nach dieser trotz generellen Verbots der Verarbeitung von personenbezogenen Daten prinzipiell möglich ist, Daten für die Kalkulation von Versicherungsprodukten zu erheben.

Lösungsvorschlag:

a) **Strukturierte Daten**:

- haben ein vorgegebenes Format, in das sich alle Informationen einordnen lassen: normalisierte Form.
- innerhalb einer relationalen Datenbank haben sie eine Zeilen- und Spaltenposition.
- sind leicht zu finden und zu bearbeiten.
- man kann sie z.B. mit SQL auslesen.
- Beispiele: Bestandsdaten in relationalen Datenbanken, etc.

Semistrukturierte Daten

- haben keine allgemeine Struktur, tragen aber einen Teil der Strukturinformationen in sich.
- sind nicht vollständig strukturiert und nicht vollständig unstrukturiert.
- Beispiel: E-Mail: Grundstruktur mit Absender, Empfänger, Betreff und weitere Informationen des Nachrichtenkopfes. Der eigentliche Inhalt ist ein strukturloser Text.

Unstrukturierte Daten:

- Informationen, die in einer nicht identifizierbaren und nicht normalisierten Datenstruktur vorliegen.
- kein Datenmodell vorhanden.
- schwer zu analysieren und zu verarbeiten.
- Beispiele: Texte, Bilder, Audio- und Videodateien, Präsentationen .

b) Mögliche Geschäftsprozesse sind:

- Gesundheitsprüfung: hier könnten Daten von Zahnarztbesuchen, Arztberichten, Röntgenbilder verwendet werden.
- Schadensmanagement: hier könnten Daten aus den eingereichten Zahnarztrechnungen ausgelesen und verwendet werden.
- Kundenmanagement: hier könnten Daten über das Anrufverhalten, bzw. Kontaktverhalten ausgewertet werden.
- Vertriebssteuerung: hier könnten Daten über die vergangenen Abschlüsse pro Region o.ä. ausgewertet werden.
- …

c) Die zentrale Rechtsquelle für Datenschutz in Europa ist die Datenschutz-Grundverordnung (DSGVO). Trotz des allgemeinen Verbots der Verarbeitung personenbezogener Daten in der DSGVO ist es prinzipiell möglich, Daten für die Kalkulation von Versicherungsprodukten zu erheben. In Arti-

kel 6 werden Ausnahmen vom Verbot aufgeführt. Für Versicherungen kommen typischerweise folgende Ausnahmen in Betracht:

a. Einwilligung der betroffenen Person.

b. Daten sind zur Vertragserfüllung nötig.

c. Daten werden benötigt, um rechtliche Verpflichtungen einzuhalten.

Aufgabe 3 [Tools & Programme] (23 Punkte)

Die folgenden Teilaufgaben a) bis d) beziehen sich auf die beiden R-Code-Snippets A und B, die weitgehend ähnliche Ergebnisse liefern.

A:

```
a0 <- data.frame(w=10:12, x=1:3, y=6:4, z=9:7)

a1 <- a0[1,3]

a2 <- a0[,"x"]

a3 <- mean(a0$z)
```

B:

```
b0 <- tibble(w=10:12, x=1:3, y=4:6, z=9:7)

b1 <- b0[1,3]

b2 <- b0[,"x"]

b3 <- b0$z %>% mean()
```

a) [6 Punkte] Welche Pakete oder Paketsammlungen werden bei A bzw. B benötigt? Führen Sie ggf. bei Paketsammlungen auch die explizit bei A bzw. B benötigten Pakete auf und beschreiben Sie in jeweils einem Satz die Funktion dieser Pakete. Wie lautet der Code zur Installation und zum Laden eines dieser Pakete? Erläutern Sie den Unterschied zwischen Installation und Laden.

b) [6 Punkte] Welche Werte und welche Datentypen (i.S. von typeof()) haben a0, a1, a2, a3 und b0, b1, b2, b3? Soweit es Unterschiede zwischen den Werten oder Datentypen der Variablen bei A und B gibt: Erläutern Sie die Ursachen der Unterschiede.

Hinweis: Benutzen Sie zur Beantwortung von Aufgabe b) bitte das der Klausur beigefügte Ergebnisformular.

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

	Snippet A		Snippet B		
	Wert /Ausgabe	**Typ (typeof)**	**Wert / Ausgabe**	**Typ (typeof)**	**Bemerkung**
a0 bzw. b0					
a1 bzw. b1					
a2 bzw. b2					
a3 bzw. b3					

Siehe Anlage

c) [4 Punkte] Erläutern Sie allgemein die Funktion von %>% in einfachen Konstellationen. Welche Vorteile kann die Verwendung von %>% haben?

Snippet A wird nun um folgende Zeilen ergänzt (zur Verdeutlichung wird hier die Definition von a0 wiederholt):

```
a0 <- data.frame(w=10:12, x=1:3, y=6:4, z=9:7)

a40 <- a0[order(a0$z),]

a41 <- a40[c("x","w")]

a41[,2] <- a41[,2] - a41[,1]

a4 <- sum(a41[,2])
```

d) [7 Punkte] Berechnen Sie den Wert von a4 und geben Sie dabei die einzelnen Zwischenschritte an. Ergänzen Sie nun Code-Snippet B, indem Sie zunächst jede dieser vier Zeilen einzeln unter bestmöglicher Nutzung des Pakets dplyr in die Ermittlung von b40, b41, etc. „übersetzen". Geben Sie alternativ eine Ermittlung von b4 unter bestmöglicher Nutzung von %>% an.

Lösungsvorschlag:

a) **A** basiert ausschließlich auf dem R-Paket base, das automatisch beim Start von R geladen wird.

B nutzt zusätzlich die R-Paketsammlung tidyverse und davon explizit das Paket tibble (für das zugrundeliegende Datenformat tibble, eine Optimierung des Datenformats data.frame) sowie das Paket magrittr (für den Pipe-Operator %>%, s. Antwort zu Teilaufgabe c). Das tidyverse ist eine Sammlung von Paketen, die speziell zur Durchführung von vollständigen Data-Science-Workflows mit strukturierten Daten entwickelt und aufeinander abgestimmt worden sind.

Code zur **Installation** des Pakets tibble: `install.packages("tibble")`

Hierdurch wird das Paket von CRAN oder einem anderen Server geholt und auf dem eigenen Rechner installiert.

Code zum **Laden** des Pakets tibble: `library(tibble)`

Hierdurch wird das Paket aus dem lokalen Installationsverzeichnis in die aktuelle R-Arbeitsumgebung geladen. Voraussetzung: Das Paket wurde zu einem früheren Zeitpunkt auf dem Rechner installiert.

Hinweis: Im Unterschied zur Installation **kann** beim Laden der **Paketname auch ohne Anführungszeichen** mitgegeben werden.

b) Die Aufrufe für a0 bis a3 bzw. b0 bis b3 liefern folgende Ergebnisse:

	Snippet A		Snippet B		
	Wert	**Typ (typeof)**	**Wert**	**Typ (typeof)**	**Bemerkung**
a0 bzw. b0	w x y z 10 1 6 9 11 2 5 8 12 3 4 7	list	w x y z 10 1 4 9 11 2 5 8 12 3 6 7	list	Die Sortierung in Spalte y ist verschieden (6:4 vs. 4:6). Data Frame (Snippet A) und Tibble (Snippet B) haben beide den Typ list.
a1 bzw. b1	6	integer	y 4	list	Das Subsetting des Data Frames liefert hier einen atomischen Vektor der Länge 1. Subsetting von Tibbles liefert immer ein Tibble (also eine list). Hinweis: Im Unterschied zu Python beginnt die Zeilen- und Spaltennummerierung bei 1 und nicht bei 0.
a2 bzw. b2	1 2 3	integer	x 1 2 3	list	Das Subsetting des Data Frames liefert hier einen atomischen Vektor der Länge 3. Subsetting von Tibbles liefert immer ein Tibble (also eine list).
a3 bzw. b3	8	double	8	double	Die Funktion mean liefert stets den Typ double zurück, auch wenn die Inputs alle den Typ integer haben und das Ergebnis ganzzahlig ist.

c) %>% ist der sog. **Pipe-Operator**, ein binärer Operator aus dem mit dem tidyverse verbundenen Paket magrittr; er wird beim Laden von tidyverse automatisch mitgeladen. Dieser Pipe-Operator ermöglicht die einfache Notation der Hintereinanderausführung mehrerer Funktionen auf „den gleichen Daten"; man spricht von der vorwärtsgerichteten Durchführung von Operationen in einer Pipeline. Genauer:

Die Kurzschreibweise `objekt %>% funktion(...)` ruft das Funktional `%>%(objekt, funktion, ...)` auf.

Wenn funktion **nur ein Argument** benötigt, dann wird bei `objekt %>% funktion` die Anweisung `funktion(objekt)` ausgeführt.

Wenn funktion **mehrere Argumente** erlaubt oder benötigt, dann wird

bei `objekt %>% funktion(arg2, arg = arg3, ...)` die Anweisung `funktion(objekt, arg2, arg = arg3, ...)` ausgeführt (d.h. per Default ist objekt das erste unbenannte Argument in funktion).

Vorteile: Der wesentliche Nutzen von %>% liegt nun in der sukzessiven Verwendung als Pipeline zur Ausführung geschachtelter Funktionsaufrufe zur Datentransformation.

Die Ausgabe des ersten Funktionsaufrufs %>% dient als Eingangsobjekt für den zweiten Funktionsaufruf %>%.

Beispiel: `Y <- X %>% f(Z1) %>% g(Z2)` entspricht der Ausführung folgender Aufrufe:

```
Y0 <- f(X, Z1)

Y <- g(Y0, Z2)
```

bzw.

```
Y <- g(f(X, Z1), Z2))
```

Im Gegensatz zum geschachtelten Funktionsaufruf führt die durch %>% entstandene Pipeline i.A. zu **selbsterklärendem Code**. Dank der hierarchischen Anordnung sind die Funktionsaufrufe in der intuitiv richtigen Reihenfolge lesbar und aufgrund der separierten schlanken Funktionsaufrufe einfacher verständlich. Vor allem werden **keine zusätzlichen Objekte zum Zwischenspeichern** von Werten oder vorab zur Definition geschachtelter Funktionen benötigt.

d) a4 = (12-3) + (11-2) + (10-1) = 9 + 9 +9 = 27.

Eine Entsprechung der einzelnen Zeilen mit dplyr könnte so aussehen:

```
b40 <- b0 %>% arrange(z)

b41 <- b40 %>% select(x,w)

b41 <- b41 %>% mutate(w=w-x)

b4 <- b41 %>% select(w) %>% sum()
```

Die Kurzfassung mit maximaler Nutzung von %>% sähe dann so aus:

```
b4 <- b0 %>%
      arrange(z) %>%
      select(x,w) %>%
      mutate(w=w-x) %>%
      select(w) %>%
      sum()
```

Anm.: Wenn der Zwischenwert b40 tatsächlich nicht von Interesse ist, kann man auch ganz auf arrange verzichten, weil das Ergebnis b4 nicht von der Sortierung der Spalte z abhängt. Dies gilt aber entsprechend bereits für a40.

Aufgabe 4 [Data Mining 1] (22 Punkte)

a) [6 Punkte] Ein Großteil des Gesamtaufwands im Data-Mining-Prozess wird der Datenvorbereitung zugeordnet. Nennen Sie sechs möglichst unterschiedliche Ursachen für die Notwendigkeit der Datenvorbereitung und geben Sie jeweils ein Beispiel aus dem Versicherungsumfeld an.

b) [6 Punkte] Erläutern Sie die Begriffe Datenselektion und Datenintegration anhand von Beispielen aus dem Versicherungsumfeld. Benennen Sie vier typische Probleme, die bei Datenselektion/-integration auftreten können.

c) [5 Punkte] Welche Arten von Daten können eine Datenbereinigung erforderlich machen? Nennen Sie fünf Arten und jeweils eine mögliche Ursache.

d) [5 Punkte] Warum gehört zur Datenvorbereitung meist eine Datentransformation? Nennen Sie vier häufig vorkommende Transformationsschritte.

Lösungsvorschlag:

Im Folgenden werden meist mehr Antwortvorschläge als gefordert genannt (nicht nur sechs Ursachen etc.). Für die volle Punktzahl ist jeweils die genannte Anzahl ausreichend. Alternative Antworten sind möglich.

a) Eine Datenvorbereitung ist insbesondere aus folgenden Gründen notwendig: Die zur Verfügung stehenden Daten

- liegen in verschiedenen Quellen – z.B. im aktuellen Bestandsführungssystem und einem Altsystem.
- haben unterschiedliche Formate – z.B. passt das Datum in unterschiedlichen Quellsystemen nicht zusammen.
- stammen von verschiedenen „Produzenten" – wenn wir nicht nur hauseigene Daten, sondern auch zugekaufte oder frei verfügbare Daten einsetzen wollen (z.B. Fondskurse).
- folgen unterschiedlichen Philosophien – z.B. Historisierung vs. „aktuellst-mögliche" Daten.
- genügen generell nicht den Qualitätsanforderungen – z.B.: Sind die Daten Wetterdaten aus einer bestimmten Internetquelle zuverlässig genug für unsere Zwecke?
- sind nicht immer repräsentativ – z.B. wenn die Studie, die verwendet werden sollte, nur mit Frauen durchgeführt wurde.
- wurden in der Regel nicht speziell für die aktuelle Data-Mining-Aufgabe erzeugt – sondern dienen z.B. primär der Bestandsführung.
- passen nicht zum vorgesehenen Data-Mining-Algorithmus – z.B. wenn dieser skalierte Werte erwartet, was bei produktiven Systemen i.d.R. nicht sinnvoll ist.

b) Bei der **Datenselektion** geht es darum, aus z.T. verschiedenen Quellen die für die aktuellen Fragestellungen erforderlichen Daten zusammenzutragen. Dies können z.B. Daten aus der eigenen Bestandsführung sein, vom GDV zur Verfügung gestellte Daten, Fondsdaten eines externen Anbieters oder Informationen aus dem Internet.

Bei der **Datenintegration** geht es darum, die im Rahmen der Datenselektion definierten Daten in ein Zielsystem zusammenzuführen, sodass diese Daten einheitlichen Regeln z.B. hinsichtlich Formatierung genügen.

Typische Probleme hierbei sind:

- Identifikation der Attribute:
  Bedeuten unterschiedlich (oder gleich) benannte Attribute das Gleiche?

- inhaltliche Unvergleichbarkeiten:
  Unterschiedliche Maßeinheiten, Währungen, Zeitzonen, Inflationseffekte; bei Kursen: Ausschüttungen und Splits.
- formale Inkonsistenzen:
  Datentypen, Datumsformat, Punkt und Komma in deutschem und amerikanischem Zahlenformat, Zeichencode.
- (scheinbare) Widersprüche:
  Unterschiedliches Geburtsdatum oder unterschiedliche Namensschreibweise in verschiedenen Quellen – Fehler oder eine andere Person?
- Verletzung referentieller Integrität:
  VN zu einem Vertrag existiert nicht.

c) Folgende Arten von Daten können eine Datenbereinigung erforderlich machen:

- Ungültige Werte – mögliche Ursache: Eine numerische Größe, die bei der Datenerfassung Strings zulässt, kann infolge eines Tippfehlers ungültig sein.
- Fehlende Daten – mögliche Ursache: Der Kunde möchte bei der Antragstellung bestimmte Informationen nicht preisgeben.
- Verrauschte Daten – mögliche Ursache: Das Telematiksystem liefert infolge eines Defekts ungenaue Messwerte.
- Ausreißer – mögliche Ursache: Durch einen Tippfehler wurde ein viel zu hoher Wert eingegeben.
- Inkonsistente Daten – mögliche Ursache: Durch einen Tippfehler passt das Geburtsdatum und das Alter nicht zusammen.

d) Daten liegen selten in der von den Algorithmen erwarteten Form vor. Häufig sind daher einige der folgenden Transformationsschritte erforderlich:

- Umwandlung kategorieller Variablen in numerische Variablen (Dummy-Variable / One-Hot-Encoding).
- Skalierung / Normalisierung / Angleichung von Maßeinheiten und Datentypen.
- Separation von Attributen (z.B. kombinierte Datums-/Zeitangabe → Stunde).
- Aggregation/Kombination von Attributen oder Datensätzen (z.B. Erfassung einzelner Klicks → täglicher Zähler).
- Datenglättung.

Aufgabe 5 [Datenverarbeitungstechnologien 1, Data Mining 1, Tools & Programme] (10 Punkte)

a) [3 Punkte] Beschreiben Sie den Begriff „reproducible Research" und nennen und beschreiben Sie die beiden Tools knitr und Jupyter Notebooks mit mindestens vier Merkmalen.

b) [4 Punkte] Nach der Entscheidung, als Entwicklungsumgebung ein Jupyter Notebook zu benutzen, beschäftigen Sie sich mit der EDA. Hierfür müssen Sie den Datensatz zunächst einlesen. Geben Sie einen Befehl in der Programmiersprache R an, um den Datensatz daten.csv (Spaltenüberschriften sind vorhanden; Die Daten sind über einen Tab getrennt und als Dezimalzeichen wird ein Komma verwendet) einzulesen.

   Beschreiben Sie im Anschluss, welche Arten von Daten in einem Datensatz vorliegen können. Gehen Sie dabei auch auf das Skalenniveau ein. Geben Sie für jede Art ein Beispiel an.

c) [3 Punkte] Nennen Sie sechs verschiedene ad-hoc-Lösungsansätze für fehlende Daten.

Lösungsvorschlag:

a) Reproduzierbare Forschung ist ein Konzept, das darauf abzielt, Ergebnisse von Analysen so zu dokumentieren, dass ihre Ableitung vollständig transparent ist. Dies kann dadurch erreicht werden, dass sowohl die benutzten Daten und als auch der verwendete Code zusammen mit den Schlussfolgerungen zugänglich gemacht werden. Man sollte sich hierbei vergegenwärtigen, dass z.B. im Pharmabereich die Rohdaten von Studien der Öffentlichkeit (und teilweise auch den Aufsichtsbehörden) nicht zugänglich gemacht werden.

   In R bietet sich dafür z.B. das R-Package knitr an. Damit können:

   i. druckbare Dokumente (html, latex, pdf, docx) aus R heraus erstellt werden.

   ii. Markdown zur Formatierung des Textes verwendet werden.

   iii. Text und Code direkt in einem Dokument bearbeitet werden.

   iv. Charts und in R-Variablen gespeicherte Datensätze als Tabellen direkt in das druckbare Dokument übernommen werden.

   v. Weitere Nennungen sind möglich.

   Jupyter Notebooks (Ursprünglich für Python entwickelte, interaktive Programmierumgebung im Webbrowser):

   i. Mischung von Code-Zellen und Markup (=Text)-Zellen.

   ii. Ausgaben aus dem Code heraus werden im Webbrowser dargestellt.

   iii. Eingebettete Graphiken möglich (z.B. Pythons matplotlib).

   iv. Client-Server-Architektur, die auch die Nutzung als Desktop-Tool am Einzelarbeitsplatz ermöglicht.

   v. Präsentationsmodus.

   vi. Direktausführung von Notebooks (sozusagen als Skripts).

   vii. Adaptiert für viele andere Sprachen/Frameworks (R, Scala, …).

   viii. Weitere Nennungen sind möglich.

b) Über R kann der Datensatz z.B. über den Befehl:

read.csv2(file, header = TRUE, sep = "\t", dec = ",")

eingelesen werden.

Prinzipiell kann zwischen quantitativen und qualitativen Merkmalen unterschieden werden. Die qualitativen Daten können dabei wie folgt unterschieden werden:

i. Nominalskala (Namen und Klassen ohne Anordnung), z.B. Farbe.

ii. Ordinalskala (Klassen und Kategorien mit Anordnung, Abstände nicht interpretierbar), z.B. Schulnote.

Die quantitativen Daten können wie folgt unterschieden werden:

i. Intervallskala (wie Ordinalskala, aber mit interpretierbaren Abständen ohne natürlichen Nullpunkt), z.B. Temperatur in Grad Celsius.

ii. Verhältnisskala (wie Intervallskala, aber mit natürlichem Nullpunkt), z.B. Schadenhöhe oder Schadenfrequenz.

c) Mögliche ad-hoc Lösungsansätze:

i. Kennzeichnung des Datensatzes als fehlerhaft.

ii. Entfernen des Datensatzes.

iii. Ignorieren / Entfernen des Attributs.

iv. Manuelle Vorgabe fehlender Werte.

v. Ersatz durch globale Konstante.

vi. Ersatz durch Modus (häufigster Wert; nicht-num. Attributen).

vii. Ersatz durch Minimum oder Maximum.

viii. Weitere Nennungen sind möglich.

Aufgabe 6 [Deep Learning 1, Tools & Programme] (35 Punkte)

a) [5 Punkte] Geben Sie die Formel für ein einfaches neuronales Netzwerk mit einem Input-Layer der Dimension 3, einem Hidden-Layer der Dimension 4 und einem Ouput-Layer der Dimension 1 an. Verwenden Sie für die Aktivierungsfunktion die Notation φ_k, für das Gewicht vom Knoten i hin zum Knoten j die Notation $\omega_{ij}^{[k]}$ sowie $a_i^{[k]}$ für das Ergebnis des Knotens i in Schicht k. Für den Bias-Term ist analog die Bezeichnung $b_i^{[k]}$ zu verwenden. Geben Sie explizit (d.h. auf Ebene der Elemente der Matrizen und Vektoren) die benötigten Berechnungen für die Forward-Propagation an (eine pro Schicht). Zudem sind die Gewichtsmatrizen anzugeben. Aus wie vielen Parametern besteht das Netzwerk?

b) [4 Punkte] Geben Sie die Formeln für die sigmoid- und ReLU-Aktivierungsfunktion an. Diskutieren Sie grundlegende Eigenschaften sowie Vor- und Nachteile.

c) [10 Punkte] Gegeben sei ein neuronales Netzwerk mit den folgenden Gewichtsmatrizen:

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

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

$$W^{[3]} = \begin{pmatrix} 0 & 2 \\ 1 & 0 \\ 0 & 0 \end{pmatrix}$$

Dabei ist zu beachten, dass die Gewichtsmatrizen in folgender Form vorliegen:

$$\begin{pmatrix} \omega_{11} & \cdots & \omega_{1n} \\ \vdots & \ddots & \vdots \\ \omega_{m1} & \cdots & \omega_{mn} \end{pmatrix}$$

Die Bias-Vektoren sind zu vernachlässigen. Skizzieren Sie das so durch diese Gewichtsmatrizen definierte Netzwerk. Führen Sie im Anschluss einen Forward-Step mit dem Input-Vektor

$$x = \begin{pmatrix} 1 \\ 1 \\ -1 \end{pmatrix}$$

durch. Aktivierungsfunktionen: Schicht 1: ReLU, Schicht 2: sigmoid, Schicht 3: ReLU. Die Berechnungen sind auf drei Stellen nach dem Komma zu runden.

d) [16 Punkte] Implementieren Sie die Ausführung eines Forward-Schritts durch ein entsprechendes R-Programm. Dabei soll nur die Netzwerkstruktur unter c) Berücksichtigung finden. Als Übergabeparameter sollen die Gewichtsmatrizen und der Input-Vektor verwendet werden.

Anmerkungen:

- Für die Gewichtsmatrizen ist anzunehmen, dass diese über die Funktion *matrix*(…) erstellt wurden.
- Es stehen nur die R Grundfunktionalitäten zur Verfügung (d.h. es kann keine weitere Library eingebunden werden).
- Die Gewichtsmatrizen werden wie Teil c) angegeben übergeben ($W^{[1]}\ bis\ W^{[3]}$).
- Benötigte Hilfsfunktionalitäten (sigmoid, ReLU) sind durch eigene Funktionen zu implementieren.

Lösungsvorschlag:

a) Mit $x = (x_1, \dots, x_3)$ ergibt sich für Layer 1:

$$a_j^{[1]} = \varphi_1\left(\sum_{i=1}^{3} \omega_{ij}^{[0]}\ a_i^{[0]} + b_j^{[0]}\right) mit\ 1 \leq j \leq 4$$

mit $a^{[0]} = x$.

Entsprechend ergibt sich für Layer 2:

$$a_1^{[2]} = \varphi_2\left(\sum_{i=1}^{4} \omega_{i1}^{[1]}\ a_i^{[1]} + b_1^{[1]}\right)$$

Gewichtsmatrizen:

$$W^{[0]} = \begin{pmatrix} \omega_{11} & \omega_{12} & \omega_{13} & \omega_{14} \\ \omega_{21} & \omega_{22} & \omega_{23} & \omega_{24} \\ \omega_{31} & \omega_{32} & \omega_{33} & \omega_{34} \end{pmatrix}$$

$$W^{[1]} = \begin{pmatrix} \omega_{11} \\ \omega_{21} \\ \omega_{31} \\ \omega_{41} \end{pmatrix}$$

Anzahl oder Parameter: 12+4+4+1=21. Hinweis: Aus Darstellungsgründen wurde der obere Index bei den Elementen der Gewichtsmatrizen weggelassen.

b) Formeln für die Aktivierungsfunktionen:

$$sigmoid(x) = \frac{1}{1 + \exp(-x)}$$

$$RELU(x) = \max(0, x)$$

Aktivierungsfunktionen sind die Elemente, die einem neuronalen Netz nichtlineare Eigenschaften verleihen. Ohne diese würden nur lineare Produkte zwischen den Eingabewerten und Gewichten berechnet werden.

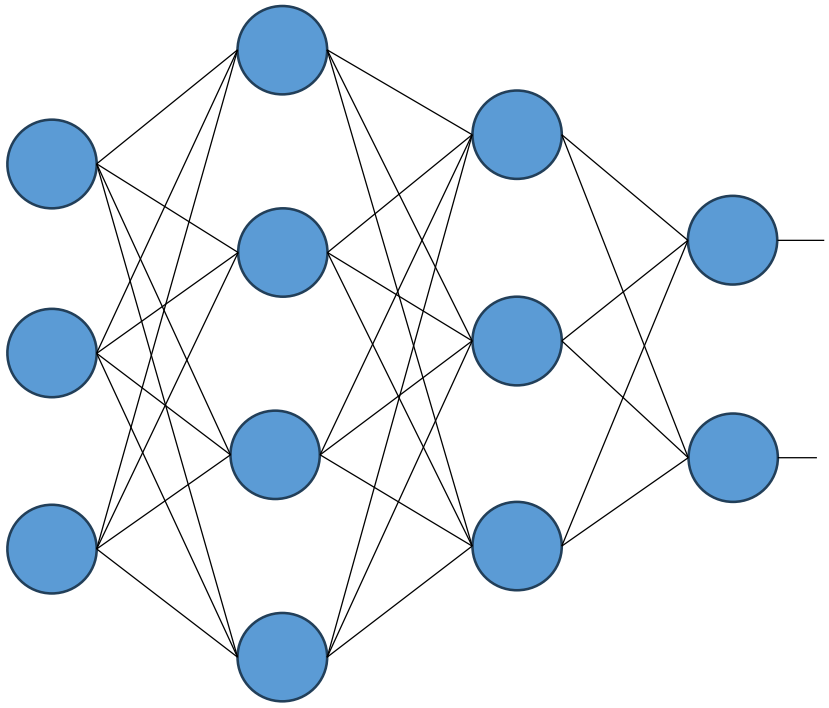
Die Sigmoidfunktion war bis vor einigen Jahren noch die häufigste Aktivierungsfunktion. Diese hat allerdings die Nachteile, dass diese bei großen bzw. kleinen Werten von x die Ableitung nahe bei Null liegt und damit den Gradienten der Verlustfunktion sehr klein machen kann. Dadurch kann es zu Problemen bei der Aktualisierung der Gewichte kommen und damit den Lernprozess erschweren bzw. verhindern. Zudem ist die Funktion nicht „null-zentriert", was das Training schwieriger und instabiler macht.

Die ReLU-Funktion ist insbesondere in den letzten Jahren populär geworden. Insbesondere hat sich gezeigt, dass sich Konvergenz des Gradientenabstiegsverfahrens in Richtung des globalen Minimums gegenüber anderen Funktionen beschleunigt. Zudem sind keine rechenaufwändigen Operationen (exp, tanh) durchzuführen. Allerdings ergeben sich durch die Tatsache, dass negative Einträge auf Null gesetzt werden auch Probleme: Einzelne Neuronen können sehr fragil werden bzw. „sterben" (d.h. deren Beitrag ist immer Null). Alternative hierzu z.B. „Leaky ReLU".

c) Auf Basis der Gewichtsmatrizen in der dargestellten Form ergibt sich ein neuronales Netzwerk folgender Struktur:

Inputschicht:	3 Knoten
Versteckte Schicht 1:	4 Knoten
Versteckte Schicht 2:	3 Knoten
Ausgabeschicht:	2 Knoten

Skizze des Netzwerkes:



Bei den Berechnungen ergeben sich die folgenden Werte:

Schritt 1:

$$a_1 = ReLU(\,W^{[1]^T} x\,) = ReLU\begin{pmatrix}-2\\1\\2\\0\end{pmatrix} = \begin{pmatrix}0\\1\\2\\0\end{pmatrix}$$

Schritt 2:

$$a_2 = \; sigmoid\Big(\,W^{[2]^T} a_1\Big) = sigmoid\begin{pmatrix}2\\2\\1\end{pmatrix} \approx \begin{pmatrix}0.88\\0.88\\0.731\end{pmatrix}$$

Schritt 3:

$$a_3 = ReLU\Big(W^{[3]^T}\, a_2\Big)) = \begin{pmatrix}0.881\\1.761\end{pmatrix}$$

d) Beispielhafte Implementierung in R:

```
W1 <- matrix(c(0,0,2,1,0,0,2,1,1,0,0,0), nrow = 3)
W2 <- matrix(c(2,0,1,-1,0,0,1,0,0,1,0,0), nrow = 4)
W3 <- matrix(c(0,1,0,2,0,0), nrow = 3)

x <- c(1,1,-1)

relu <- function(x) {max(0,x)}
sigmoid <- function(x) {1/(1+exp(-x))}

FFNN_forward <- function(W1,W2,W3,x) {
        z_1 <- t(W1) %*% x
        a_1 <- sapply(z_1, FUN=relu)

        z_2 <- t(W2) %*% a_1
        a_2 <- sapply(z_2, FUN=sigmoid)

        z_3 <- t(W3) %*% a_2
        a_3 <- sapply(z_3, FUN=relu)
}

r <- FFNN_forward(W1,W2,W3,x)
print(r)
```

Aufgabe 7 [Unüberwachtes Lernen 1] (25 Punkte)

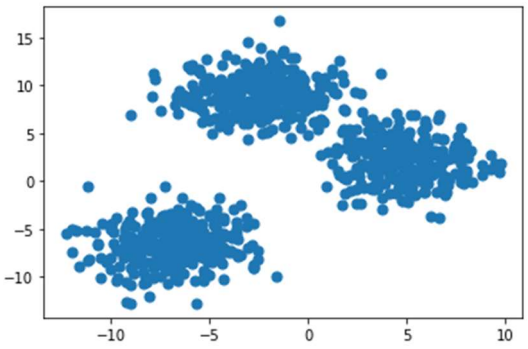
a) [6 Punkte] Beschreiben Sie den k-Means-Algorithmus in wenigen Sätzen.

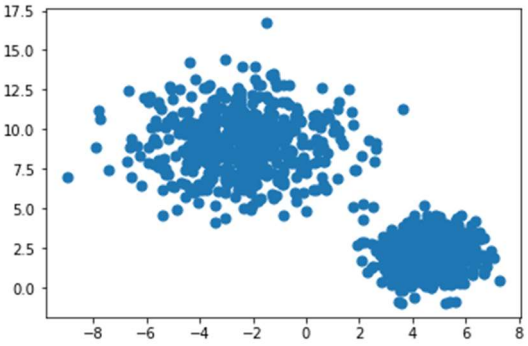
b) [6 Punkte] Geben Sie zwei Beispiele aus dem Versicherungsumfeld an, bei denen man Clusteringverfahren anwenden kann, und den jeweiligen Nutzen.

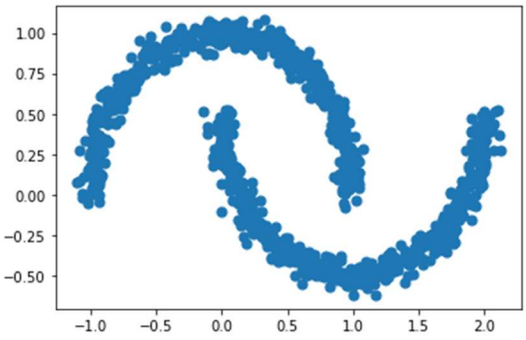
c) [9 Punkte] Entscheiden Sie bei den folgenden Plots zunächst, wie viele Cluster k jeweils vorliegen. Skizzieren Sie auf der Basis dieses von Ihnen gewählten k jeweils ein mögliches Ergebnis des k-Means-Algorithmus.

Bei welchem Plot liefert der Algorithmus kein sinnvolles Ergebnis?

Nennen Sie eine Eigenschaft, die Cluster haben sollten, damit der Algorithmus sinnvolle Ergebnisse liefert.

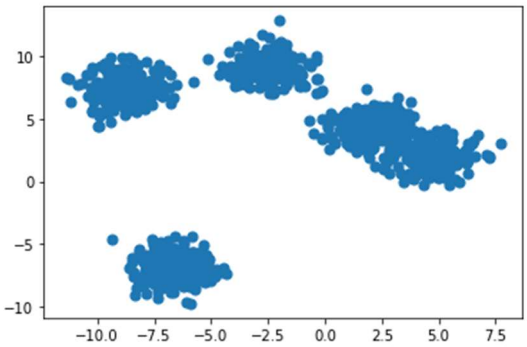




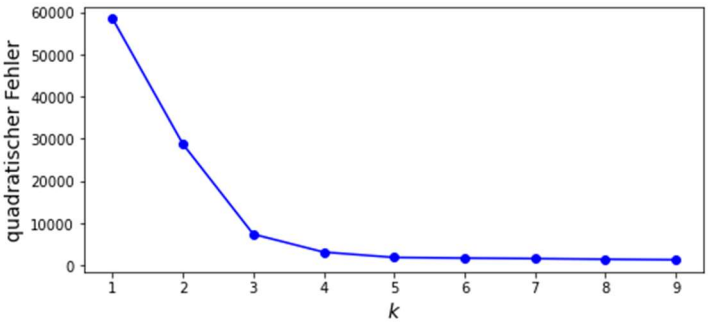


d) [4 Punkte]

Entscheiden Sie, analog zu Teil c), wie viele Cluster Sie bei diesem Datensatz vermuten:



Sie führen für diesen Datensatz den k-Means-Algorithmus für ein bis neun Cluster durch. Anschließend plotten Sie die jeweiligen Summen der quadratischen Abweichungen und bekommen den folgenden Verlauf:



Wie viele Cluster würden Sie anhand dieses Plots auswählen (mit Begründung)?

Lösungsvorschlag:

a)

Die Datenpunkte werden zunächst zufällig einem der k Cluster zugeordnet. (1 Punkt)

Solange bis sich die Cluster nicht mehr ändern, werden folgende Schritte ausgeführt: (5 Punkte)

- Berechnung der Clusterzentren als (koordinatenweise) Durchschnittswerte aller Punkte im jeweiligen Cluster.
- Zuordnung der Datenpunkte zu dem Cluster, bei dem das Clusterzentrum den geringsten euklidischen Abstand aufweist.

b) Beispiel 1: Clusteringverfahren können zur Segmentierung der Versicherungsnehmer benutzt werden. Die Segmentierung kann dann für maßgeschneiderte Bestandsaktionen benutzt werden.

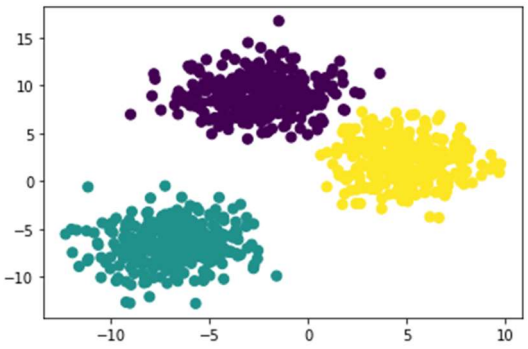
Beispiel 2: Für Bestandsprojektionen können Clusteringverfahren benutzt werden, um den Bestand zu verdichten und die Laufzeit der Projektion zu reduzieren.

Beispiel 3: Bei Bestandsmigrationen können die Verträge geclustert werden, um eine repräsentative Auswahl als Testbestand festzulegen.

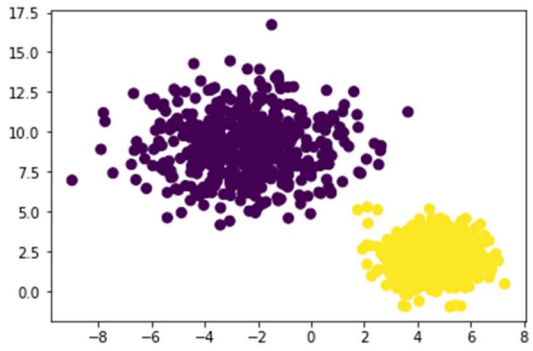
Die Angabe von zwei Beispielen genügt für die volle Punktzahl.

c)

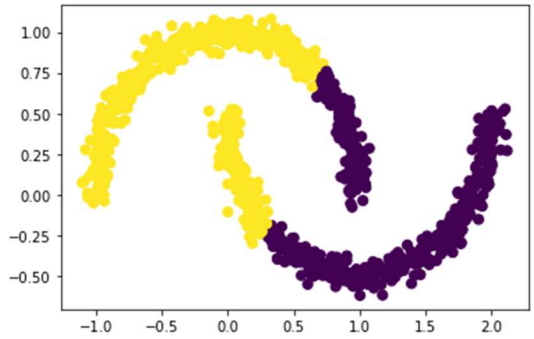
Beim ersten Plot liegen drei Cluster vor. k-Means könnte folgendes Clustering liefern: (2 Punkte)



Beim zweiten Plot liegen zwei Cluster vor. k-Means könnte folgendes Clustering liefern: (2 Punkte)



Beim dritten Plot liegen zwei Cluster vor. k-Means könnte folgendes Clustering liefern: (2 Punkte)



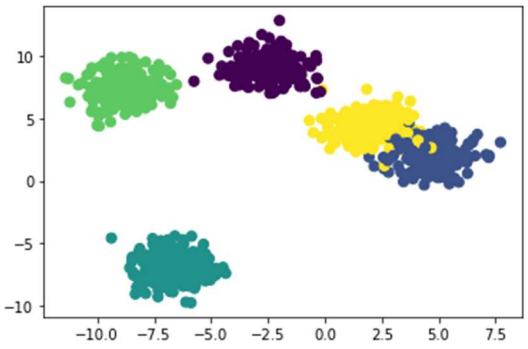
Offensichtlich liefert k-Means beim dritten Plot kein sinnvolles Ergebnis. (1 Punkt) Damit k-Means sinnvolle Ergebnisse liefert, sollten die einzelnen Cluster näherungsweise kreisförmig (bzw. kugelförmig in höheren Dimensionen) sein. Diese Eigenschaft ist beim dritten Plot verletzt. (2 Punkte)

d)

Es sind vier Cluster zu erkennen. Tatsächlich wurden für das Beispiel fünf kreisförmige Cluster erzeugt. Richtige Antworten sind daher vier oder fünf. (2 Punkte)

Anhand des Plots der quadratischen Fehler wählen wir auch vier Cluster. Der quadratische Fehler sinkt bis zu vier Clustern deutlich. Bei mehr als vier Clustern sinkt der quadratische Fehler nur noch unwesentlich. (2 Punkte)

Hinweis: Der zugrundeliegende Datensatz sieht wie folgt aus (nicht Teil der Lösung):

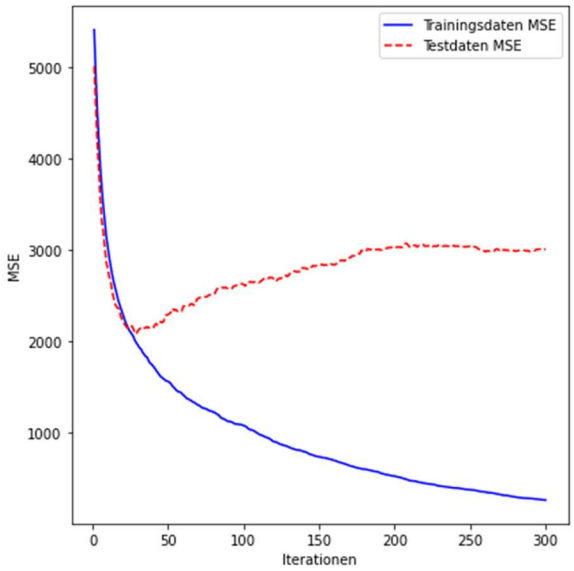

10
5
0
−5
−10
−10.0
−7.5
−5.0
−2.5
0.0
2.5
5.0
7.5

Aufgabe 8 [Analytics 1, Überwachtes Lernen 1] [20 Punkte]

a) [5 Punkte] Beschreiben Sie das Boosting Verfahren.

b) [5 Punkte] Die folgende Grafik zeigt, wie sich der MSE auf den Trainings- und Testdaten in Abhängigkeit von der Anzahl der Iterationen des benutzten Boosting-Verfahrens entwickelt.

Welches Phänomen sehen Sie? Wie würden Sie aufgrund der Grafik die Anzahl der Bäume bei dem Boosting-Verfahren festlegen?



c) [5 Punkte] Wie unterscheiden sich Bagging und Boosting? Kann es bei Entscheidungsbäumen in Verbindung mit Bagging zu Overfitting kommen? Begründen Sie bitte Ihre Antwort.

d) [5 Punkte] Warum ist es sinnvoll, neben einem Trainings- und einem Testdatensatz noch einen Validierungsdatensatz zu erzeugen?

Lösungsvorschlag:

a) Es wird ein additives Modell erstellt. Dazu werden rekursiv Entscheidungsbäume gelernt. Der nächste Entscheidungsbaum wird basierend auf den Residuen des bereits trainierten Modells, d.h. auf der Differenz zwischen Vorhersage und tatsächlichen Werten, trainiert. Die Summe der Entscheidungsbäume wird zur Vorhersage genutzt.

b) Hier ist deutlich ein Overfitting zu sehen. Bis zur ungefähr 25. Iteration sinken die MSEs auf den Trainings- und Testdaten und sind fast gleich. Ab dann sinkt der MSE auf den Trainingsdaten weiter stark, während der MSE auf den Testdaten wieder ansteigt. Aufgrund der Grafik sollte man die Anzahl Bäume auf ungefähr 25 festlegen.

c) Beim Boosting werden die einzelnen Lerner rekursiv ermittelt und diese dann additiv kombiniert. Beim Bagging werden stochastisch unabhängig mehrere Lerner ermittelt und diese dann durch Mittelung (bei Regression) oder Mehrheitsentscheidung (bei Klassifikation) kombiniert. Beim Bagging kommt es in der Regel nicht zum Overfitting, da die einzelnen Lerner in der Theorie alle gleich stark und stochastisch unabhängig sind, sodass die Vorhersagequalität mit der Anzahl der benutzten Bäume steigt.

d) Trainingsdaten werden benutzt, um das Modell zu trainieren. Validierungsdaten werden benutzt, um die richtigen Parameter zu bestimmen. Testdaten werden benutzt, um nach Ende der Modellerstellung das Modell zu evaluieren. Die Testdaten in dieser Aufgabe sind also eigentlich Validierungsdaten.

DAV

DEUTSCHE
AKTUARVEREINIGUNG e.V.

Written examination in

Actuarial Data Science Basic

According to Prüfungsordnung 5
of Deutsche Aktuarvereinigung e. V.

on the 24th of May 2024

Notes:

- A 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 been given for completeness. The exam consists of 33 pages.
- All answers must be justified and the solution to arithmetic problems must be clear.

Members of the examination commission:

Axel Kiermaier, Dr. René Külheim, Dr. Jonas Offtermatt, Dr. Felix Spangenberg

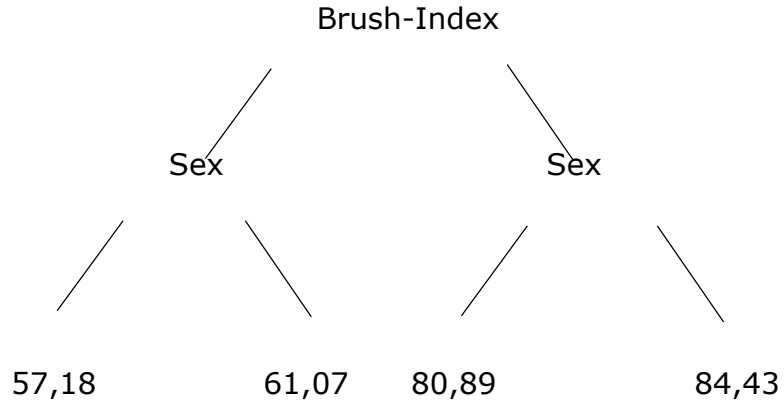
Exercise 1 [Data management 1, Information processing in insurance companies, Supervised learning 1] (27 Points)

In the table below you find claims data for the Pay-as-you-brush product from your company. This product is a supplementary dental insurance where the price is adjusted according to a brush index. The brush index is determined via the feedback from a smart toothbrush.

Customer_ID	Policy_Number	Brush_Index	Sex	Age	Premium	Claims
CI1	11.84579	Low	Female	22	88.44	84.43
CI2	11.95096	Low	Male	47	95.70	80.89
CI3	11.28109	High	Other	64	115.81	61.07
CI4	11.77030	High	Female	42	78.10	57.11
CI5	11.50399	High	Other	55	107.50	43.05
CI6	11.26674	Low	Other	57	87.33	38.76
CI7	11.39113	Low	Other	54	80.85	61.66
CI8	11.32006	Low	Female	58	118.68	81.94
CI9	11.41755	Low	Female	51	108.54	72.54
CI10	11.42011	Medium	Male	44	88.37	57.18

a) [4 Points] In order to be able to process the data automatically, it should be saved in a machine-readable format. Write down the first three lines of the above table in csv format.

b) [4 Points] The amount of damage is to be estimated using a tree method. In a first test, only the features Brush_Index and Sex, as well as the data records with the customer ID CI1, CI2, CI3 and CI10 were used to generate an incomplete decision tree. Sketch of the generated tree:



Transfer the sketch of the tree to your solution sheets and add the corresponding characteristic criteria to the lines.

c) [9 Points] Assume that the above data is available in an R dataset named insurance_data. The following R code is given:

```
# Create data for the bar plot
plot_data <- insurance_data %>%
  group_by(Sex, Brush_Index) %>%
  summarize(Mean_Claims = mean(Claims))

# Plot the bar plot
ggplot(plot_data, aes(x = Sex, y = Mean_Claims, fill = Brush_Index)) +
  geom_col(position = "dodge") +
  labs(title = "Claims by Sex and Brush Index", y = "Mean Claims")
```

Draw the diagram generated by this code.

Hints: The command print(`plot_data`) generates:

```
# A tibble: 6 × 3
# Groups:   Sex [3]
  Sex    Brush_Index Mean_Claims
  <chr>  <fct>             <dbl>
1 Female Low                79.6
2 Female High               57.1
3 Male   Low                80.9
4 Male   Medium             57.2
5 Other  Low                50.2
6 Other  High               52.1
```

`Position = “dodge”` causes bars to be printed next to each other in a bar chart.

d) [10 Points] Your colleagues want to prepare the data in Excel and store the Excel file in the local file system. You argue in favour of an R notebook and storing it in a central code repository. Explain why this approach better fulfils the general requirements for software. In doing so, address the criteria of audit acceptability, maintainability, scalability, flexibility, and affordability.

Solution proposal:

a) There are several ways to save the data in a csv file. For the full score, it is necessary to also save the line names in the csv.

Valid would be:

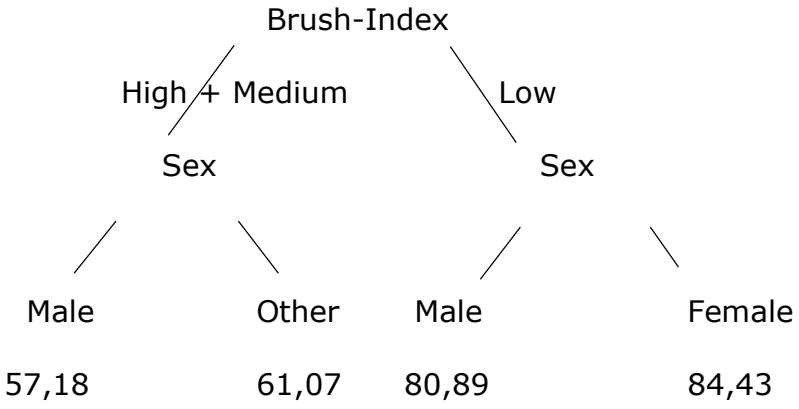
```
"","Customer_ID","Policy_Number","Brush_Index","Sex","Age","Premium",
"Claims"
"1","CI1","11.84579","Low","Female",22,88.44,84.43
"2","CI2","11.95096","Low","Male",47,95.7,80.89
```

Or:

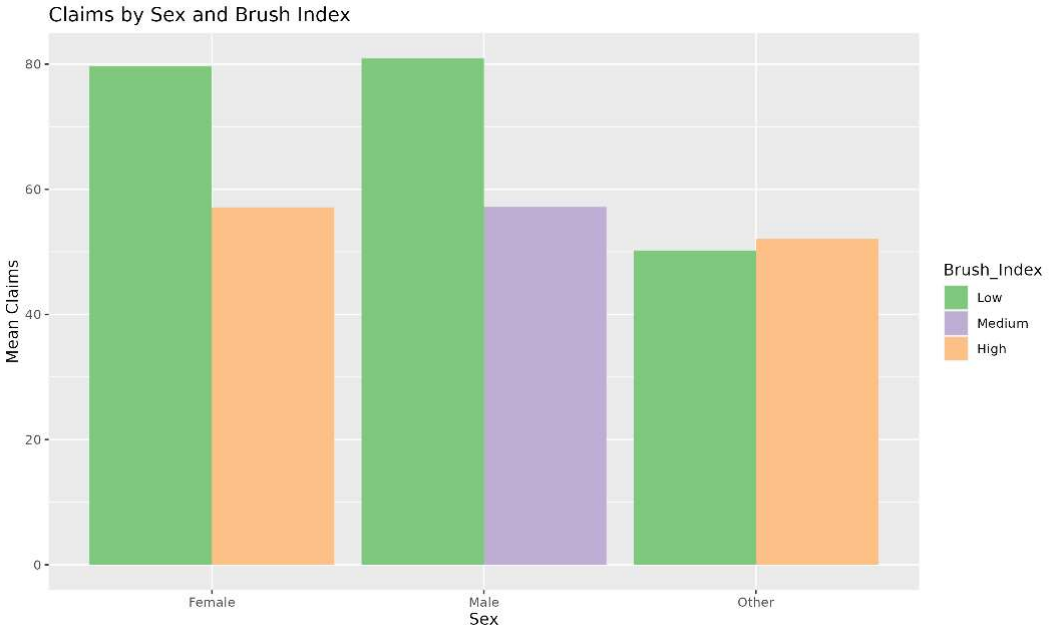
```
"";"Customer_ID";"Policy_Number";"Brush_Index";"Sex";"Age";"Premium";
"Claims"
"1";"CI1";"11.84579";"Low";"Female";22;88,44;84,43
"2";"CI2";"11.95096";"Low";"Male";47;95,7;80,89
```

It is also valid to drop the first column with the numbering.

b)



c) The code creates the following plot:



d) Here are some reasons why using an R notebook in a centralized code repository better meets the general requirements for software compared to using Excel files:

Audit Accountability:

R notebooks are text-based and can be easily stored and managed in version control systems such as Git. This enables accurate tracking of changes, team collaboration and the restoration of previous versions.

Maintanability:

You can document and comment the code in R notebooks, which improves the readability and maintainability of the code. This also makes it easier for other team members to collaborate and understand the code.

Scalability:

R enables the automation of tasks and the processing of large amounts of data. Unlike Excel, in R you are not restricted to manual steps and limited data sizes.

Flexibility:

With R-Notebooks, you can integrate various R packages and external data sources to perform sophisticated analyses and visualizations. The flexibility of R allows you to customize your analysis to specific requirements.

Affordability:

R is an open source platform and available free of charge, unlike Excel, which often requires paid licenses. This lowers the cost of the software and allows access to powerful analysis tools at no additional cost.

Overall, R notebooks provide better structure and organization for data-based tasks and facilitate adherence to best software development practices. This is especially important when it comes to performing data analysis in enterprise or research environments where transparency, traceability and collaboration are key.

Exercise 2 [Data protection 1, Data management 1, Information processing in insurance companies] (18 Points)

Your company would like to relaunch an existing insurance product. Machine learning methods are to be used for the first time to calculate premiums more accurately.

a) [9 Points] In order to be able to calculate the premiums, the existing data is to be expanded to include unstructured data. Explain the difference between structured, unstructured and semi-structured data and give an example for each data type.

b) [3 Points] Name three insurance company business processes from which you could obtain further data for calculating premiums for the above product.

c) [6 Points] Name the central legal source for data protection within Europe and explain why it is possible in principle to collect data for the calculation of insurance products despite a general ban on the processing of personal data.

Solution proposal:

a) **Structured data**:

- has a predefined format into which all information can be classified: normalized form.
- has a row and column position within a relational database.
- is easy to find and edit.
- can be read with SQL, for example.
- Examples: Inventory data in relational databases, etc.

Semi-structured data

- has no general structure, but carry part of the structural information within them.
- is not completely structured and not completely unstructured.
- Example: e-mail: Basic structure with sender, recipient, subject and other information in the message header. The actual content is a structureless text.

Unstructured data:

- Information that is available in a non-identifiable and non-normalized data structure.
- No data model available.
- difficult to analyse and process.
- Examples: Texts, images, audio and video files, presentations.

b) Possible business processes are:

- Health check: data from dentist visits, medical reports, x-rays could be used here.
- Claims management: data from submitted dental bills could be read and used here.
- Customer management: data on call or contact behaviour could be evaluated here.
- Sales management: data on past contracts per region or similar could be evaluated here.
- ...

c) The central legal source for data protection in Europe is the General Data Protection Regulation (GDPR). Despite the general prohibition on the processing of personal data in the GDPR, it is in principle possible to collect data for the calculation of insurance products. Article 6 lists exceptions to the prohibition. The following exceptions typically apply to insurance companies:

a. Consent of the data subject

b. Data is necessary for the performance of the contract

c. Data is required to comply with legal obligations.

Exercise 3 [Tools & Programs] (23 Points)

The following subtasks a) to d) refer to the two R code snippets A and B, which provide largely similar results.

A:

```
a0 <- data.frame(w=10:12, x=1:3, y=6:4, z=9:7)

a1 <- a0[1,3]

a2 <- a0[,"x"]

a3 <- mean(a0$z)
```

B:

```
b0 <- tibble(w=10:12, x=1:3, y=4:6, z=9:7)

b1 <- b0[1,3]

b2 <- b0[,"x"]

b3 <- b0$z %>% mean()
```

a) [6 Points] Which packages or package collections are required for A or B? For package collections, also list the packages explicitly required for A or B and describe the function of these packages in one sentence. What is the code for installing and loading one of these packages? Explain the difference between installation and loading.

b) [6 Points] Which values and which data types (in the sense of `typeof()`) do a0, a1, a2, a3 and b0, b1, b2, b3 have? If there are differences between the values or data types of the variables in A and B: Explain the causes of the differences.

Note: For answering task b), please use the result form attached to the exam.

Make sure to submit the page you have worked on additionally!

	Snippet A		Snippet B		
	Value / Output	Type (typeof)	Value / Output	Type (typeof)	Remarks
a0 resp. b0					
a1 resp. b1					
a2 resp. b2					
a3 resp. b3					

See attachment

c) [4 Points] Explain the general function of %>% in simple constellations. What advantages can the use of %>% have?

Snippet A is now supplemented by the following lines (for clarification, the definition of a0 is repeated here):

```
a0 <- data.frame(w=10:12, x=1:3, y=6:4, z=9:7)

a40 <- a0[order(a0$z),]

a41 <- a40[c("x","w")]

a41[,2] <- a41[,2] - a41[,1]

a4 <- sum(a41[,2])
```

d) [7 Points] Calculate the value of a4 and specify the individual intermediate steps. Now complete code snippet B by first "translating" each of these four lines individually into the determination of b40, b41, etc. using the package dplyr in the best possible way. Alternatively, specify a determination of b4 with the best possible use of %>%.

Solution proposal:

a) **A** is based exclusively on the R package base, which is automatically loaded when R is started.

B additionally uses the R package collection tidyverse, specifically the package tibble (for the underlying data format tibble, an optimization of the data format data.frame) as well as the package magrittr (for the pipe operator %>%, see the answer to subtask c). The tidyverse is a collection of packages that have been specially developed and coordinated for the implementation of complete data science workflows with structured data.

Code for installing the tibble package: `install.packages("tibble")`

> This fetches the package from CRAN or another server and installs it on your own computer.

Code for loading the tibble package: `library(tibble)`

> This loads the package from the local installation directory into the current R working environment. Prerequisite: The package was installed on the computer at an earlier time.

Note: In contrast to the installation, the package name can also be entered without quotation marks when loading.

b) The calls for a0 to a3 or b0 to b3 provide the following results:

	Snippet A		Snippet B		
	Value / Output	**Type (typeof)**	**Value / Output**	**Type (typeof)**	**Remarks**
a0 resp. b0	w x y z 10 1 6 9 11 2 5 8 12 3 4 7	list	w x y z 10 1 4 9 11 2 5 8 12 3 6 7	list	The sorting in column y is different (6:4 vs. 4:6). Data Frame (snippet A) and Tibble (snippet B) are both lists.
a1 resp. b1	6	integer	y 4	list	The subsetting of the data frame gives the atomic vector, with length 1. Subsetting a tibble always returns a tibble (i.e., a list). Note: Unlike Python, row and column numbering starts at 1 instead of 0.
a2 resp. b2	1 2 3	integer	x 1 2 3	list	The subsetting of the data frame gives the atomic vector, with length 3. Subsetting a tibble always returns a tibble (i.e., a list).
a3 resp. b3	8	double	8	double	The function mean returns a double, regardless of the type of input.

c) %>% is the so-called pipe operator, a binary operator from the magrittr package associated with the tidyverse; it is automatically loaded when tidyverse is loaded. This pipe operator enables the simple notation of the successive execution of several functions on "the same data"; this is referred to as the forward execution of operations in a pipeline. More precisely:

The shorthand notation `object %>% function(...)` calls the functional

`%>%(object, function, ...)`.

If `function` requires only one argument, then the instruction function(object) is executed for object %>% function.

If function allows or requires several arguments, then for object %>% function(arg2, arg = arg3, ...) the statement function(object, arg2, arg = arg3, ...) is executed (i.e. by de-fault object is the first unnamed argument in function).

Advantages: The main benefit of %>% now lies in its successive use as a pipeline for executing nested function calls for data transformation.

The output of the first function call %>% serves as the input object for the second function call %>%.

Example: `Y <- X %>% f(Z1) %>% g(Z2)` corresponds to the execution of the following calls:

```
Y0 <- f(X, Z1)

Y <- g(Y0, Z2)
```

resp.

```
Y <- g(f(X, Z1), Z2))
```

In contrast to the nested function call, the pipeline created by %>% generally leads to self-explanatory code. Thanks to the hierarchical arrangement, the function calls can be read in the intuitively correct order and are easier to understand due to the separated lean function calls. Above all, no additional objects are required to store values temporarily or to define nested functions in advance.

d) a4 = (12-3) + (11-2) + (10-1) = 9 + 9 +9 = 27.

An equivalent of the individual lines with dplyr could look like this:

```
b40 <- b0 %>% arrange(z)

b41 <- b40 %>% select(x,w)

b41 <- b41 %>% mutate(w=w-x)

b4 <- b41 %>% select(w) %>% sum()
```

The short version with maximum utilization of %>% would then look like this:

```
b4 <- b0 %>%
      arrange(z) %>%
      select(x,w) %>%
      mutate(w=w-x) %>%
      select(w) %>%
      sum()
```

Note: If the intermediate value b40 is actually not of interest, you can also dispense with arrange completely, because the result b4 does not depend on the sorting of column z. However, this also applies to a40.

Exercise 4 [Data Mining 1] (22 Points)

a) [6 Points] A large part of the total effort in the data mining process is allocated to data preparation. Name six different reasons for the necessity of data preparation and give an example from the insurance sector for each reason.

b) [6 Points] Explain the terms data selection and data integration using examples from the insurance sector. Name four typical problems that can occur with data selection/integration.

c) [5 Points] What types of data may require data cleansing? Name five types and one possible cause for each.

d) [5 Points] Why does data preparation usually include a data transformation? Name four frequently occurring transformation steps.

Solution proposal:

In the following, there are usually more suggested answers than required (not just six causes etc.). For the full score, the number mentioned is sufficient. Alternative answers are possible.

a) Data preparation is particularly necessary for the following reasons: The available data

- is in different sources - e.g., in the current inventory management system and a legacy system.
- has different formats - e.g., the date in different source systems does not match.
- originates from different "producers" - if we want to use not only in-house data, but also purchased or freely available data (e.g., fund prices).
- follows different philosophies - e.g., historicization vs. "as up to date as possible" data.
- generally, does not meet the quality requirements - e.g.: Is the data weather data from a particular internet source reliable enough for our purposes?
- is not always representative - e.g., if the study to be used was only conducted with women.
- was usually not generated specifically for the current data mining task - but is primarily used for inventory management, for example.
- does not match the intended data mining algorithm - e.g. if the algorithm expects scaled values, which is generally not appropriate for productive systems.

b) **Data selection** involves compiling the data required for the current issues from various sources. This can be data from the company's own portfolio management, data provided by the GDV, fund data from an external provider or information from the Internet.

The aim of **data integration** is to merge the data defined in the data selection process into a target system so that this data complies with standardized rules, e.g., with regard to formatting.

Typical problems are:

- Identification of attributes: Do differently (or identically) named attributes mean the same thing?

- Incomparability of content: Different units of measurement, currencies, time zones, inflation effects; for prices: distributions and splits.
- Formal inconsistencies: Data types, date format, dot and comma in German and American number format, character code.
- (Apparent) contradictions: Different date of birth or different name spelling in different sources - error or another person?
- Violation of referential integrity: VN for a contract does not exist.

c) The following types of data may require data cleansing:

- Invalid values - possible cause: A numerical value that allows strings during data entry may be invalid due to a typing error.
- Missing data - possible cause: The customer does not want to disclose certain information when submitting an application.
- Noisy data - possible cause: The telematics system provides inaccurate measured values due to a defect.
- Outlier - possible cause: A much too high value was entered due to a typing error.
- Inconsistent data - possible cause: Due to a typing error, the date of birth and age do not match.

d) Data is rarely available in the form expected by the algorithms. Therefore, some of the following transformation steps are often necessary:

- Conversion of categorical variables into numerical variables (Dummy-Variable / One-Hot-Encoding).
- Scaling / normalization / alignment of measurement units and data types.
- Separation of attributes (e.g., combined date/time or hour).
- Aggregation/combination of attributes or data sets (e.g., recording of individual clicks or daily counters).
- Data smoothing.

Exercise 5 [Data Processing Technologies 1, Data Mining 1, Tools & Programs] (10 Points)

a) [3 Points] Describe the term "reproducible research" and name and describe the two tools knitr and Jupyter Notebooks with at least four characteristics.

b) [4 Points] Once you have decided to use a Jupyter Notebook as your development environment, you can start working with the EDA. To do this, you must first read in the data set. Enter a command in the programming language R to read in the data set daten.csv (column headers are available; the data is separated by a tab and a comma is used as the decimal point).

   Then describe which types of data can be present in a data set. Also describe the scale level. Give an example for each type.

c) [3 Points] Name six different ad-hoc solutions for missing data.

Solution proposal:

a) Reproducible research is a concept that aims to document the results of analyses in such a way that their derivation is completely transparent. This can be achieved by making both the data used and the code used accessible together with the conclusions. It should be borne in mind that in the pharmaceutical sector, for example, the raw data from studies are not made available to the public (and in some cases also to the regulatory authorities).

   In R, for example, the R package knitr can be used for this. This can be used to:

   i. printable documents (html, latex, pdf, docx) can be created from R.

   ii. Markdown can be used to format the text.

   iii. Text and code can be edited directly in a document.

   iv. Charts and data sets stored in R variables can be transferred directly into the printable document as tables.

   v. Further entries are possible.

   Jupyter Notebooks (originally developed for Python, interactive programming environment in the web browser):

   i. Mixture of code cells and markup (=text) cells.

   ii. Output from the code is displayed in the web browser.

   iii. Embedded graphics possible (e.g., Python's matplotlib).

   iv. Client-server architecture, which also enables use as a desktop tool at a single workstation.

   v. Presentation mode.

   vi. Direct execution of notebooks (as scripts, so to speak).

   vii. Adapted for many other languages/frameworks (R, Scala, ...).

   viii. Further mentions are possible.

b) In R the data set could be read, with the following command:

read.csv2(file, header = TRUE, sep = "\t", dec = ",")

In principle, a distinction can be made between quantitative and qualitative characteristics. Qualitative data can be differentiated as follows:

i. Nominal scale (names and classes without order), e.g., color.

ii. Ordinal scale (classes and categories with arrangement, distances cannot be interpreted), e.g., school grade.

The quantitative data can be broken down as follows:

i. Interval scale (like ordinal scale, but with interpretable intervals without natural zero), e.g., temperature in degrees Celsius.

ii. Ratio scale (like interval scale, but with natural zero point), e.g., damage level or damage frequency.

c) Possible ad-hoc solutions:

i. Marking the data record as incorrect.

ii. Removal of the data record.

iii. Ignoring / removing the attribute.

iv. Manual specification of missing values.

v. Replacement by global constant.

vi. Replacement by mode (most frequent value; non-num. attributes).

vii. Replacement by minimum or maximum.

viii. Further entries are possible.

Exercise 6 [Deep Learning 1, Tools & Programs] (35 Points)

a) [5 Points] Enter the formula for a simple neural network with an input layer of dimension 3, a hidden layer of dimension 4 and an output layer of dimension 1. Use the following notation for the activation function φ_k, for the weight of the node i to node j the notation $\omega_{ij}^{[k]}$ and $a_i^{[k]}$ for the result of node i in layer k. For the bias term, the designation $b_i^{[k]}$ is to be used. Specify explicitly (i.e., at the level of the elements of the matrices and vectors) the required calculations for the forward propagation (one per layer). The weight matrices must also be specified. How many parameters does the network consist of?

b) [4 Points] State the formulas for the sigmoid and ReLU activation function. Discuss basic properties as well as advantages and disadvantages.

c) [10 Points] Below you'll find a neural network with the following weight matrices:

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

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

$$W^{[3]} = \begin{pmatrix} 0 & 2 \\ 1 & 0 \\ 0 & 0 \end{pmatrix}$$

It should be noted that the weight matrices are in the following form:

$$\begin{pmatrix} \omega_{11} & \cdots & \omega_{1n} \\ \vdots & \ddots & \vdots \\ \omega_{m1} & \cdots & \omega_{mn} \end{pmatrix}$$

The bias vectors are to be neglected. Sketch the network defined by these weight matrices. Then carry out a forward step with the input vector:

$$x = \begin{pmatrix} 1 \\ 1 \\ -1 \end{pmatrix}$$

Activation functions: Layer 1: ReLU, Layer 2: sigmoid, Layer 3: ReLU. The calculations must be rounded to three decimal places.

d) [16 Points] Implement the execution of a forward step using a corresponding R program. Only the network structure under c) should be considered. The weight matrices and the input vector should be used as transfer parameters.

Remarks:

- For the weight matrices, it can be assumed that these were created using the `matrix`(...) function.
- Only the R basic functionalities are available (i.e., no further library can be integrated).
- The weight matrices are transferred as specified in part c) ($W^{[1]}$ *to* $W^{[3]}$).
- Required auxiliary functions (sigmoid, ReLU) must be implemented using your own functions.

Solution proposal:

a) With $x = (x_1, \ldots, x_3)$ Layer 1 results in:

$$a_j^{[1]} = \varphi_1\left(\sum_{i=1}^{3} \omega_{ij}^{[0]}\ a_i^{[0]} + b_j^{[0]}\right) with\ 1 \leq j \leq 4$$

with $a^{[0]} = x$.

Accordingly, the following results are obtained for layer 2:

$$a_1^{[2]} = \varphi_2\left(\sum_{i=1}^{4} \omega_{i1}^{[1]}\ a_i^{[1]} + b_1^{[1]}\right)$$

Weight matrices:

$$W^{[0]} = \begin{pmatrix} \omega_{11} & \omega_{12} & \omega_{13} & \omega_{14} \\ \omega_{21} & \omega_{22} & \omega_{23} & \omega_{24} \\ \omega_{31} & \omega_{32} & \omega_{33} & \omega_{34} \end{pmatrix}$$

$$W^{[1]} = \begin{pmatrix} \omega_{11} \\ \omega_{21} \\ \omega_{31} \\ \omega_{41} \end{pmatrix}$$

Number of parameters: 12+4+4+1=21. Note: For display reasons, the upper index has been omitted from the elements of the weight matrices.

b) Formulas for the activation functions:

$$sigmoid(x) = \frac{1}{1 + \exp(-x)}$$

$$RELU(x) = \max(0, x)$$

Activation functions are the elements that give a neural network non-linear properties. Without these, only linear products between the input values and weights would be calculated.

Until a few years ago, the sigmoid function was still the most common activation function. However, this has the disadvantage that the derivative is close to zero for large or small values of x and can therefore make the gradient of the loss function very small. This can lead to problems when updating the weights and thus complicate or prevent the learning process. In addition, the function is not "zero-centered", which makes training more difficult and unstable.

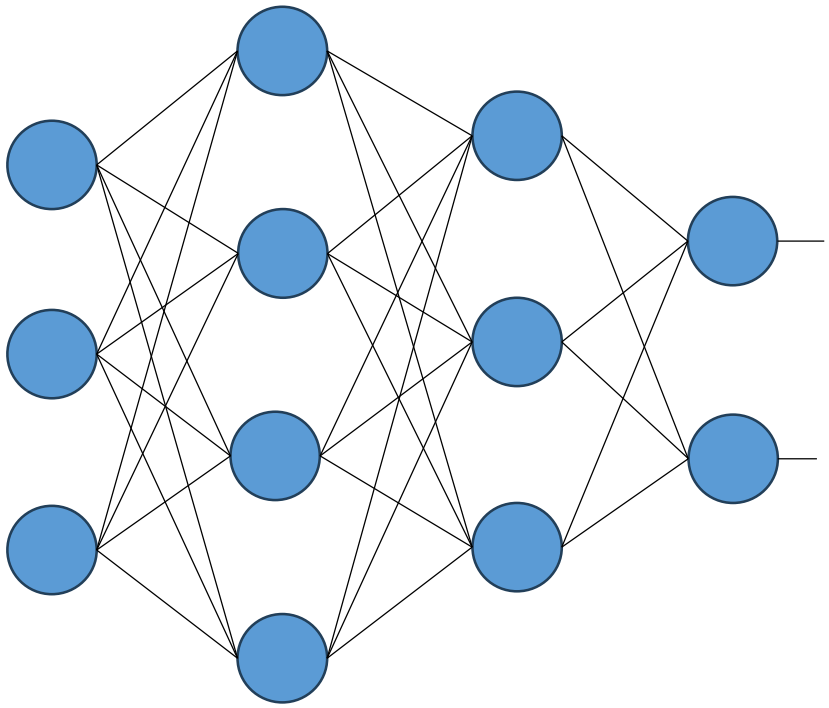
The ReLU function has become particularly popular in recent years. In particular, it has been shown that convergence of the gradient descent meth-

od towards the global minimum accelerates compared to other functions. In addition, no computationally expensive operations (exp, tanh) need to be performed. However, the fact that negative entries are set to zero also causes problems: individual neurons can become very fragile or "die" (i.e. their contribution is always zero). Alternative to this e.g. "Leaky ReLU".

c) Based on the weight matrices in the form shown, a neural network with the following structure results:

Inputlayer:	3 Nodes
Hidden Layer 1:	4 Nodes
Hidden Layer 2:	3 Nodes
Outputlayer:	2 Nodes

Sketch of the network:



The calculations result in the following values:

Step 1:

$$a_1 = ReLU(\,W^{[1]^T} x\,) = ReLU\begin{pmatrix} -2 \\ 1 \\ 2 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ 2 \\ 0 \end{pmatrix}$$

Step 2:

$$a_2 = sigmoid\left(W^{[2]^T} a_1\right) = sigmoid\begin{pmatrix}2\\2\\1\end{pmatrix} \approx \begin{pmatrix}0.88\\0.88\\0.731\end{pmatrix}$$

Step 3:

$$a_3 = ReLU\left(W^{[3]^T} a_2\right)) = \begin{pmatrix}0.881\\1.761\end{pmatrix}$$

d) Exemplary implementation in R:

```
W1 <- matrix(c(0,0,2,1,0,0,2,1,1,0,0,0), nrow = 3)
W2 <- matrix(c(2,0,1,-1,0,0,1,0,0,1,0,0), nrow = 4)
W3 <- matrix(c(0,1,0,2,0,0), nrow = 3)

x <- c(1,1,-1)

relu <- function(x) {max(0,x)}
sigmoid <- function(x) {1/(1+exp(-x))}
softmax <- function(x) {
        exp(x) / sum(exp(x))
}

FFNN_forward <- function(W1, W2, W3, x) {
        z_1 <- t(W1) %*% x
        a_1 <- sapply(z_1, FUN=relu)

        z_2 <- t(W2) %*% a_1
        a_2 <- sapply(z_2, FUN=sigmoid)

        z_3 <- t(W3) %*% a_2
        a_3 <- sapply(z_3, FUN=relu)
}

r <- FFNN_forward(W1, W2, W3, x)
print(r)
```

Exercise 7 [Unsupervised Learning 1] (25 Points)

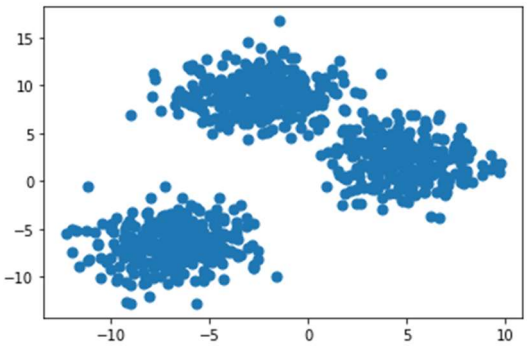
a) [6 Points] Describe the k-Means algorithm in a few sentences.

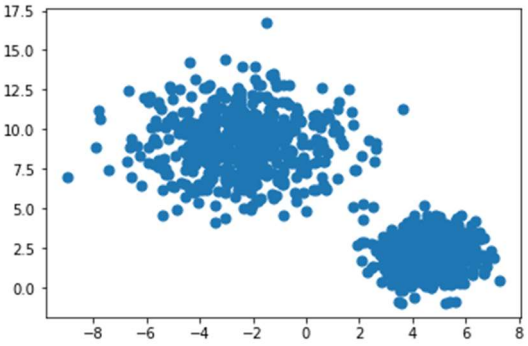
b) [6 Points] Name two examples from the insurance environment where clustering methods can be used and the respective benefits.

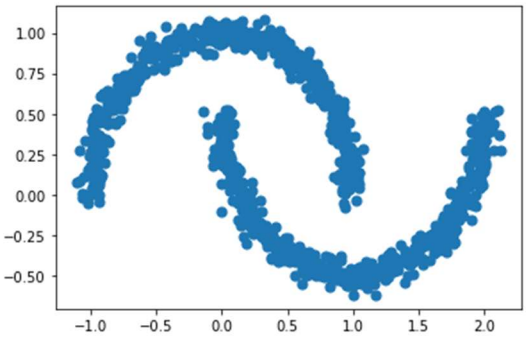
c) [9 Points] For the following plots, first decide how many clusters k are present in each case. Based on the k you have chosen, sketch a possible result of the k-means algorithm in each case.

For which plot does the algorithm not provide a meaningful result?

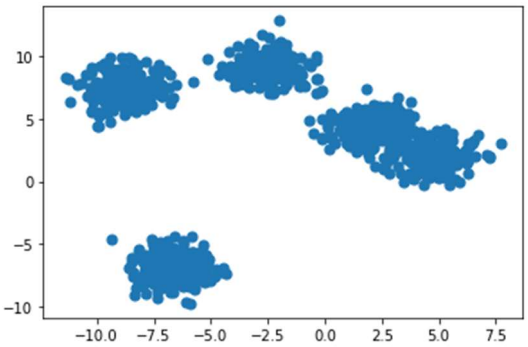
Name one property that clusters should have in order for the algorithm to deliver meaningful results.



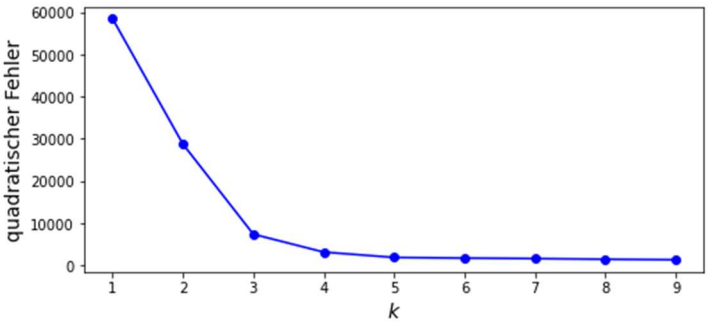




d) [4 Points] Analogous to part c), decide how many clusters you assume for this data set:



You perform the k-means algorithm for this data set for one to nine clusters. You then plot the respective sums of the quadratic deviations and obtain the following curve:



How many clusters would you select based on this plot (with reasons)?

Solution proposal:

a) The DataPoints are first randomly assigned to one of the k clusters. (1 point)

The following steps are carried out until the clusters no longer change: (5 points)

- Calculation of the cluster centers as (coordinate-wise) average values of all points in the respective cluster.
- Assignment of the data points to the cluster in which the cluster center has the smallest Euclidean distance.

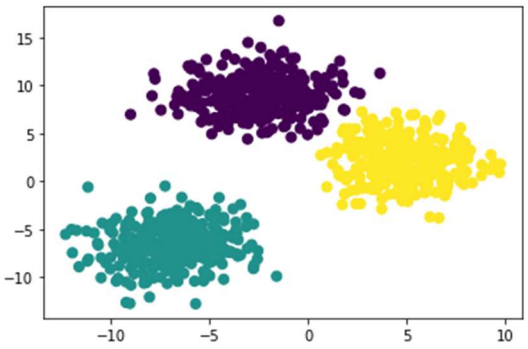
b) Example 1: Clustering methods can be used to segment policyholders. The segmentation can then be used for customized policy actions.

Example 2: For portfolio projections, clustering methods can be used to condense the portfolio and reduce the duration of the projection.

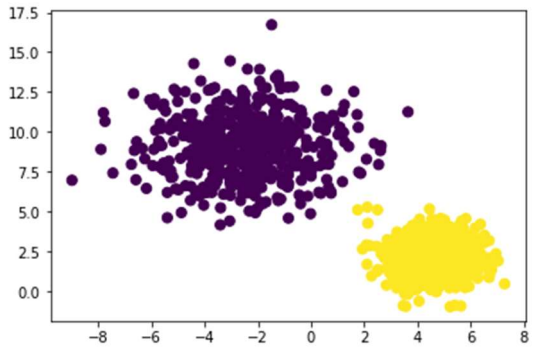
Example 3: For portfolio migrations, the contracts can be clustered to define a representative selection as a test portfolio.

The specification of two examples is sufficient for the full score.

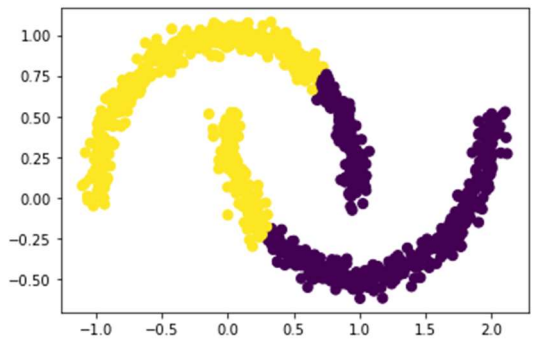
c) The first plot contains three clusters. k-Means could provide the following clustering: (2 Points)



The second plot has two clusters. k-Means could provide the following clustering: (2 Points)



The third plot contains two clusters. k-Means could provide the following clustering: (2 Points)

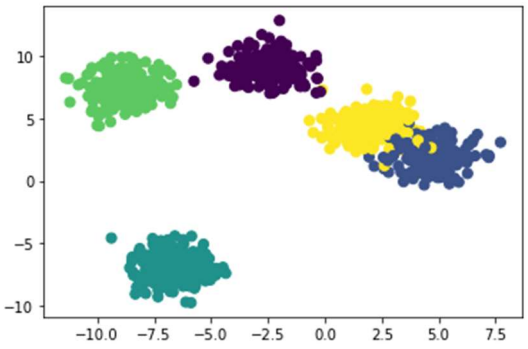


Obviously, k-Means does not provide a meaningful result for the third plot. (1 point) In order for k-Means to provide meaningful results, the individual clusters should be approximately circular (or spherical in higher dimensions). This property is violated in the third plot. (2 Points)

d) Four clusters can be recognized. In fact, five circular clusters were generated for the example. The correct answers are therefore four or five. (2 points)

Based on the plot of the squared error, we also select four clusters. The squared error decreases significantly up to four clusters. With more than four clusters, the squared error decreases only insignificantly. (2 points)

Note: The underlying data set looks as follows (not part of the solution):

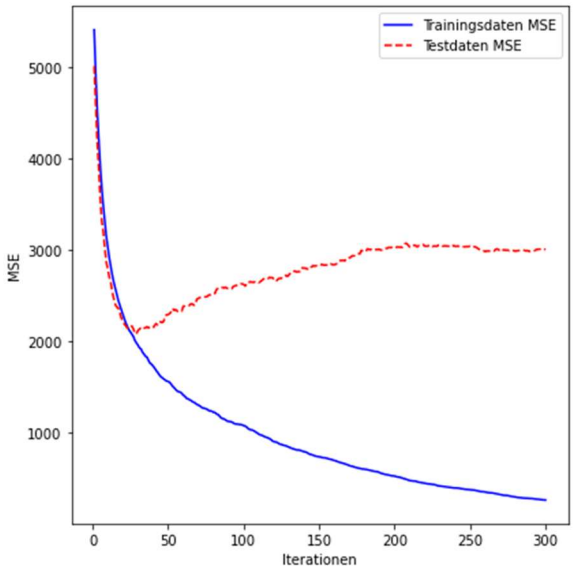

10
5
0
−5
−10
−10.0
−7.5
−5.0
−2.5
0.0
2.5
5.0
7.5

Exercise 8 [Analytics 1, Supervised Learning 1] (20 Points)

a) [5 Points] Describe the *Boosting* process.

b) [5 Points] The following graph shows how the MSE develops on the training and test data as a function of the number of iterations of the boosting method used.

What phenomenon do you see? Based on the graph, how would you determine the number of trees in the boosting process?



c) [5 Points] What is the difference between bagging and boosting? Can overfitting occur with decision trees in conjunction with bagging? Please explain your answer.

d) [5 Points] Why is it useful to create a validation data set in addition to a training data set and a test data set?

Solution proposal:

a) An additive model is created. For this purpose, decision trees are learned recursively. The next decision tree is trained based on the results of the already trained model, i.e., on the difference between prediction and actual values. The sum of the decision trees is used for prediction.

b) An overfitting can be clearly seen here. Up to about the 25th iteration, the MSEs on the training and test data decrease and are almost equal. From then on, the MSE on the training data continues to fall sharply, while the MSE on the test data rises again. Based on the graph, the number of trees should be set to approximately 25.

c) In boosting, the individual learners are determined recursively and then combined additively. In bagging, several learners are determined stochastically independently and then combined by averaging (in the case of regression) or majority decision (in the case of classification). Bagging generally does not lead to overfitting, as the individual learners are in theory all equally strong and stochastically independent, so that the prediction quality increases with the number of trees used.

d) Training data is used to train the model. Validation data is used to determine the correct parameters. Test data is used to evaluate the model after it has been created. So the test data in this task is actually validation data.