DAV

DEUTSCHE
AKTUARVEREINIGUNG e.V.

Schriftliche Prüfung im Spezialwissen

Actuarial Data Science Advanced

gemäß Prüfungsordnung 4.1
der Deutschen Aktuarvereinigung e. V.

am 27.10.2023

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

Mitglieder der Prüfungskommission:

Axel Kiermaier, Dr. René Külheim, Prof. Dr. Jonas Offtermatt, Tobias Renner, Dr. Felix Spangenberg

Aufgabe 1 [6.1 Datenmanagement 2, 5.1 Gesellschaftliches Umfeld & Ethik 2, 5.2 Datenschutz 2, 8.3 Innovative Produkte 2] (36 Punkte)

Sie sind Teamleiter der Data Science-Gruppe der Pfefferminzia Versicherung. Die Pfefferminzia hat in den letzten Jahren keine neuen Produkte eingeführt und eher schlechte Unternehmenszahlen ausgewiesen. Trotzdem ist es gelungen, einen neuen Vertriebsvorstand zu verpflichten. Sie erläutern ihm am ersten Tag die Lage aufbauend auf folgenden Unternehmenszahlen:

a) (12 Punkte) Visualisieren Sie (gerne kreativ) die Unternehmenskennzahlen aus Tabelle 1, so dass dem Vorstand die Lage des Unternehmens schnell eindrücklich wird. Verwenden Sie hierfür mind. 5 der vorgegebenen Kennzahlen und verfassen Sie zu Ihre(m/n) Schaubild(ern) jeweils eine kurze Erläuterung.

Ihr neuer Vorstand ist erschüttert. So schlecht hat er sich die Lage nicht vorgestellt. Um den Schock zu verarbeiten, zieht er die Quelle der Daten in Frage. Als Sie antworten, diese seien aus dem eigenen Data Warehouse, schaut Sie der Vorstand nur fragend an.

b) (7 Punkte) Erläutern Sie ihrem Vorstand das Konzept eines Data Warehouse und gehen Sie dabei insbesondere auf die Vorteile des Sternschemas ein.

Der Vorstand schluckt, beruhigt sich und Sie aber damit, dass Versicherungen ja schon seit hunderten von Jahren bestehen und so schnell nicht vom Markt verschwinden werden. Sie widersprechen:

c) (4,5 Punkte) Nennen Sie drei Beispiele für disruptive Veränderungen in verschiedenen Branchen.

So langsam dämmert es Ihrem neuen Vorstand, dass neue Ideen gefragt sind. Er schaut Sie auffordernd an:

d) (4,5 Punkte) Nennen Sie drei Beispiele (andere als pay-as-you-live Produkten) für innovative Produktneueinführungen der Versicherungsbranche aus den letzten 5 Jahren.

Ihr Vorstand ist begeistert und will sofort ein pay-as-you-live-Produkt im Bereich Leben einführen.

e) (8 Punkte) Weisen Sie Ihren Vorstand auf die Risiken solcher Produkte hin. Nennen und erläutern Sie hierfür kurz vier verschiedene Risiken, die bei Einführung eines solchen Produktes bedacht werden sollten.

DAV

DEUTSCHE AKTUARVEREINIGUNG e.V.

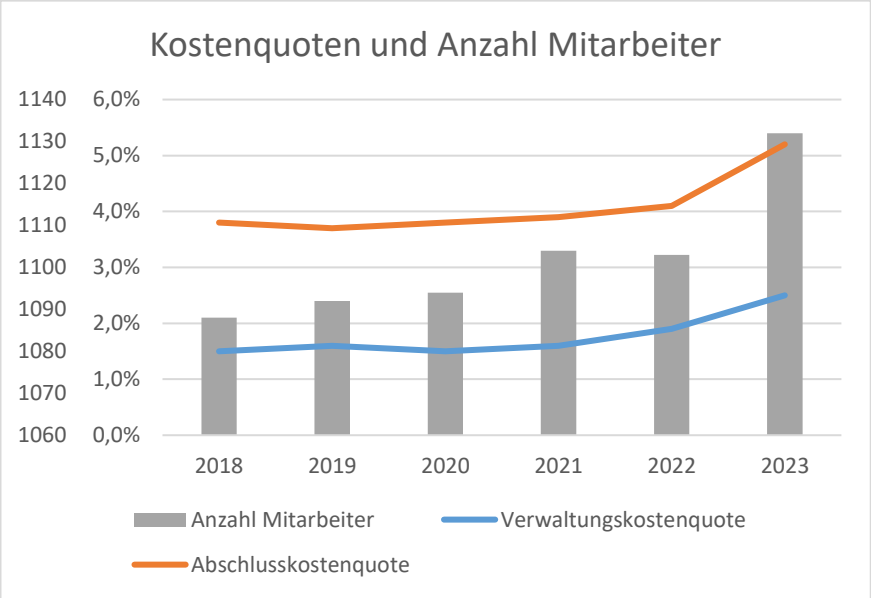
Kennzahlen Übersicht Pfefferminzia

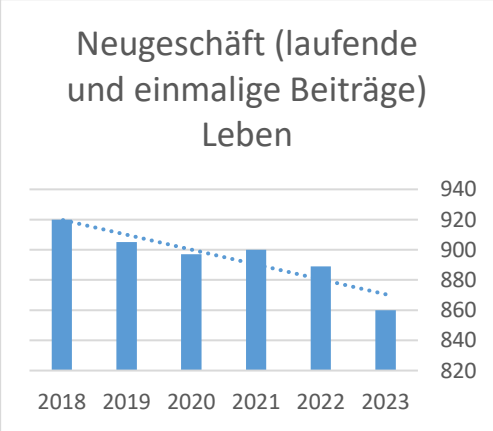
		2023	2022	2021	2020	2019	2018
Pfefferminzia Leben							
Neugeschäft (laufende und einmalige Beiträge)	Mio €	860	889	900	897	905	920
Veränderung	%	-3,3%	-1,2%	0,3%	-0,9%	-1,6%	1,8%
Versicherungsbestand (laufender Beitrag ein Jahr)	Mio €	1988	1998	2033	2045	2123	2053
Veränderung	%	-0,5%	-1,7%	-0,6%	-3,7%	3,4%	4,0%
Stornoquote	%	3,2%	2,9%	2,5%	2,0%	2,1%	2,0%
Kapitalanlagen	Mio €	27.780	29.710	30.193	30.150	31.003	30.324
Veränderung	%	3,2%	3,2%	3,2%	3,2%	3,2%	3,2%
Nettoverzinsung	%	2,6%	2,1%	3,6%	3,5%	3,2%	3,6%
Verwaltungskostenquote	%	1,8%	1,7%	1,6%	1,5%	1,6%	1,5%
Abschlusskostenquote	%	4,3%	4,1%	3,9%	3,8%	3,7%	3,8%
Eigenkapital	Mio €	212	234	254	255	257	261
Eigenkapitalquote	‰	32	35,1	39,8	39,9	40,2	41,2
Pfefferminzia Sach							
Anzahl Verträge		4.444.321	4.495.567	4.799.123	4.912.354	5.011.675	5.123.514
Veränderung	%	-1%	-6%	-2%	-2%	-2%	-2%
gebuchte Bruttobeiträge	Mio €	957	1022	997	1011	1033	1053
Veränderung	%	-6%	3%	-1%	-2%	-2%	-1%
Leistungsausgaben	Mio €	1203	1354	1051	1143	987	993
Veränderung	%	-11,2%	28,8%	-8,0%	15,8%	-0,6%	0,4%
Schaden-Kostenquote	%	102,9	104,5	101,7	102,3	101,01	99,6
Pfefferminzia Konzern							
Eigenkapital	Mio €	423	454	468	465	472	481
Veränderung		-6,8%	-3,0%	0,6%	-1,5%	-1,9%	-0,8%
Anzahl Mitarbeiter		1132	1103	1104	1094	1092	1088
Net Promoter Score		-16	-4	4	15	20	25
Anzahl Beschwerden bei Ombudsmann		12	6	5	2	3	1

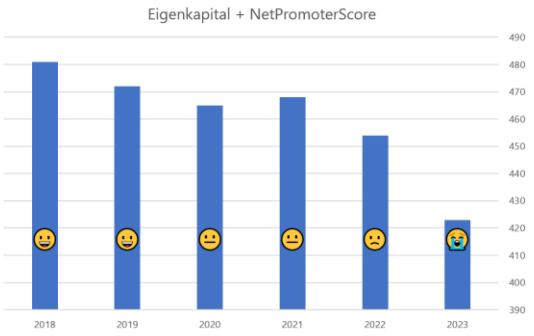
Lösungsvorschlag:

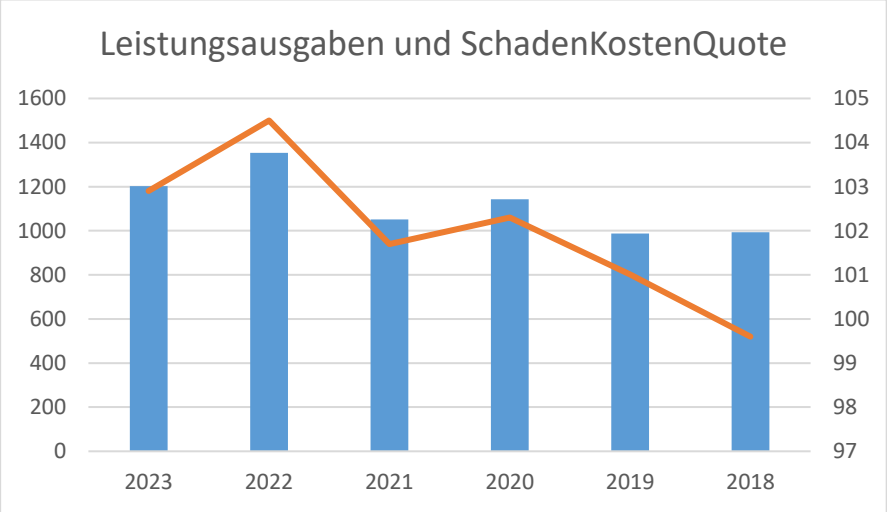
a) Es gibt viele Möglichkeiten die Lage zu visualisieren. Da eigentlich bei allen Kennzahlen die Werte schlechter werden, eignen sich voraussichtlich Balken-Diagramme am ehesten (diese werden auch von den meisten Menschen verstanden).

Beispiele:

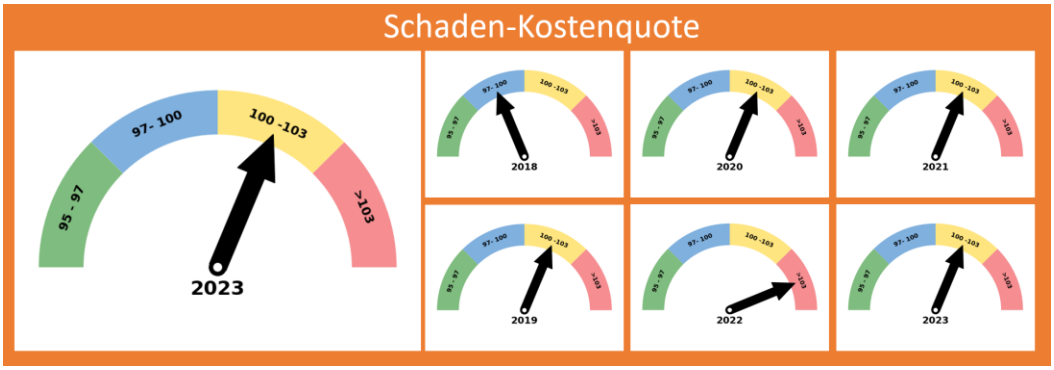








Kreativere Darstellungen:



b) Ein Data Warehouse ist eine von operativen Systemen separierte Datenbank mit folgenden Eigenschaften:

- Aus operativen Systemen werden zyklisch (z.B. real time, täglich) Daten zusammengetragen, vereinheitlicht, geordnet, verdichtet und dauerhaft archiviert.

- Ziel ist die Verbesserung der unternehmensinternen Informationsversorgung (Wissensmanagement) und es dient damit der Unterstützung strategischer Entscheidungen.

- Der Anwender kann nur lesend zugreifen.

Beim Sternschema findet sich im Zentrum eine Faktentabelle, die sternförmig mit Dimensionstabellen verknüpft ist. Die Faktentabelle enthält als Attribute die ökonomischen Kennzahlen, die im Report ausgewiesen werden sollen. Der Primärschlüssel setzt sich aus den einzelnen Primärschlüsseln

der Dimensionstabellen zusammen. In der Regel verletzen die Dimensionstabellen die dritte Normalform. Dadurch kann eine tiefverzweigte Datenstruktur vermieden werden.

Vorteile: Das Datenmodell ist einfach und intuitiv. Die Abfragegeschwindigkeit ist deutlich verbessert, weil mehrstufige Joins vermieden werden.

c) Beispiele (die Liste ist nicht allumfassend): [1,5 Punkte pro Beispiel]

   a) Quelle und Neckermann wurden von Amazon verdrängt

   b) Bagger mit Hydraulik haben Bagger mit Seilzügen verdrängt

   c) Die Wikipedia hat den Brockhaus obsolet gemacht

   d) Musikstreaming hat den Verkauf von physischen Tonträgern fast vollständig zum Erliegen gebracht

   e) Ebenso das Videostreaming die Videotheken und DVD-/Videoplayer

   f) Das Automobil verdrängte die Kutsche

   g) Digitalkameras verdrängten die klassischen Kameras (und werden jetzt von Handykameras verdrängt)

   h) …

d) Beispiele: [1,5 Punkte pro Beispiel]

   a) P2P-Versicherung von Friendsurance

   b) Telematik-Tarife im Bereich Kfz

   c) Flugverspätungsversicherung auf Basis der Blockchain (Fizzy von Axa)

   d) Schadenregulierung per Foto-Übermittlung und Auswertung durch Machine-Learning (StateFarm)

   e) Beam Dental günstigere Zahnzusatzversicherung wenn mit smarter Zahnbürste die Zähne geputzt werden (Kein Spaß, https://www.beambenefits.com/dental).

   f) …

e) Folgende Risiken treten bei neuen datengetriebenen Produkten auf [2 Punkte pro Risiko]:

    a) Reputationsrisiko: Risiko als Datenkrake wahrgenommen zu werden

    b) Technologie-Risiko: Die Verwaltung von größeren Datenmengen benötigt eine bessere technische Infrastruktur

    c) Reputationsrisiko bei Datenverlust (Hacker, etc.)

    d) Andere Betrugsrisiken: Hund bekommt Armbanduhr um, Telematik-Tracker fährt in einem anderem Auto mit,…

    e) …

Aufgabe 2 [Regressions- und Clustermethoden 2 (7.1), Data Mining 2 (8.1), Analytics 2 (8.2)] *(38 Punkte)*

Im Rahmen der Kfz-Versicherung haben Sie die Aufgabe erhalten, ein Prognosemodell zu erstellen, welches vorhersagt, ob ein beschädigtes Fahrzeug einen Totalschaden hat. Dieses Prognosemodell soll in der Schadenregulierung verwendet werden, um gezielt bei Totalschäden einen Totalschadenprozess einzuleiten. Auf Basis von historischen Daten wissen Sie, dass ca. 40 % der beschädigten Fahrzeuge einen Totalschaden aufweisen.

In der folgenden Tabelle sind zehn exemplarische Datensätze aus zwei Prognosemodellen dargestellt:

Fahrzeug_id	Fahrzeug-marke	Erstzulas-sungsjahr	Totalschaden-prognose Modell 1	Totalschaden-prognose Modell 2	Fahrzeugzustand
100001	BMW	2023	3%	23%	Kein Totalschaden
100002	VW	2007	38%	89%	Kein Totalschaden
100003	AUDI	1989	88%	83%	Totalschaden
100004	VW	2006	67%	53%	Totalschaden
100005	VW	2006	21%	95%	Totalschaden
100006	BMW	1999	63%	65%	Kein Totalschaden
100007	VW	2001	7%	38%	Kein Totalschaden
100008	AUDI	2012	9%	48%	Kein Totalschaden
100009	VW	2018	83%	40%	Totalschaden
100010	BMW	2022	5%	47%	Kein Totalschaden

a) [7 Punkte] Für die Prognosemodelle wird ein Lift-Chart und ein ROC-Chart erstellt ausgewählt. Beschreiben Sie den Nutzen und die Zielsetzung eines Lift-Charts und eines ROC-Charts und benennen Sie die Prozessphase im CRISP-DM, in dem die Charts erstellt und verwendet werden.

b) [14 Punkte] Beschreiben Sie die Schritte zur Erstellung eines Lift-Chart und eines ROC-Chart. Erstellen Sie für die Datensätze aus der Aufgabenstellung ein Lift-Chart für das Prognosemodell 1. Berechnen Sie hierzu die Datenpunkte und zeichnen Sie das Lift-Chart. Berechnen Sie für die Datensätze aus der Aufgabenstellung einen exemplarischen Datenpunkt für das ROC-Chart.

c) [6 Punkte] Ein Kollege schlägt Ihnen vor, die Methode „Bagging" anzuwenden. Beschreiben Sie die Methode „Bagging" und erläutern Sie die Vorteile der Methode.

d) [11 Punkte] Zusätzlich zu den zwei Prognosemodellen soll ein Random Forest mit einer Boosting-Methode erstellt werden. Hierbei soll die Methode AdaBoost (Adaptive Boosting) verwendet werden. Beschreiben Sie die verschiedenen Schritte zur Erzeugung des Random Forest mit der Boosting-Methode.

Lösungsvorschlag:

a) Lift-Charts und ROC-Charts werden in der Evaluierungsphase im CRISP-DM verwendet mit dem Ziel, die Performance von einem oder mehreren Prognosemodellen zu bewerten und die Performance von verschiedenen Modellen zu vergleichen.

In einem Lift-Chart wird der Nutzen von einem oder mehreren Klassifikationsmodellen dargestellt. Hierbei wird dargestellt, um welchen Faktor ein Prognosemodell besser darin ist, ein Ereignis korrekt vorherzusagen, im Vergleich ohne Prognosemodell.

In einem ROC-Chart wird die Performance / Prognosegüte von einem oder mehreren Klassifikationsmodellen für alle Klassifikationsschwellwerte (Cut-Off Werte) dargestellt, mit dem Ziel der Bewertung der Modellgüte. Hierbei wird die True Positive Rate (Sensitivity) gegen die False Positive Rate (1 - Specificity) dargestellt.

b) Für die Erstellung des Lift-Charts sind, aufbauend auf den prognostizierten Daten, die nachfolgenden Schritte notwendig. Die Erstellung eines Lift-Charts sollte auf dem Test-Datensatz erfolgen:

1. Sortiere die Datensätze absteigend nach der prognostizierten Wahrscheinlichkeit
2. Teile die Datensätze in gleichgroße Gruppen auf (häufig erfolgt eine Aufteile in Dezile / zehn Gruppen)
3. Ermittlung des Lift_i bis zur jeweiligen Gruppe i. Hierbei ist der Lift definiert als:

$$Lift_i = \frac{Anz.\,der\ Erg.\,mit\ dem\ Prognosemodell\ für\ die\ Datensätze\ der\ ersten\ i\ Gruppen}{Anz.\,der\ zufälligen\ Anzahl\ von\ Erg.\,für\ die\ Datensätze\ der\ ersten\ i\ Gruppen}$$

Für die exemplarischen Datensätze erfolgt die Berechnung und Erstellung des Lift-Charts mit dem Prognosemodell 1 wie folgt:

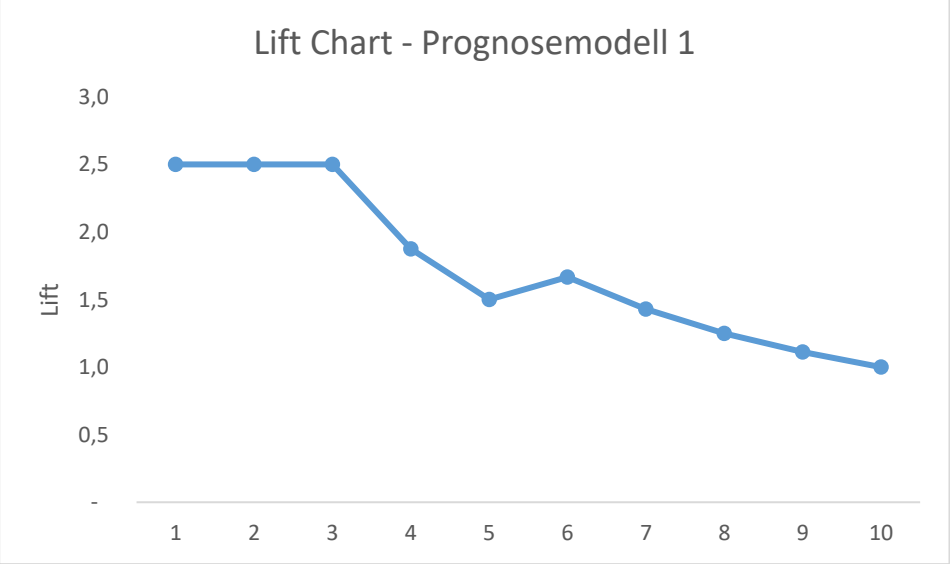
1. Sortierung der Datensätze:

Totalschadenprognose Modell 1	**Fahrzeugzustand**
88%	Totalschaden
83%	Totalschaden
67%	Totalschaden
63%	Kein Totalschaden
38%	Kein Totalschaden
21%	Totalschaden

9%	Kein Totalschaden
7%	Kein Totalschaden
5%	Kein Totalschaden
3%	Kein Totalschaden

2. Jeder der zehn Datensätze umfasst eine eigene Gruppe
3. Aus der Aufgabenstellung ist zu entnehmen, dass 40 % der Fahrzeuge ein Totalschaden sind. Hieraus und aus den Prognosewerten ergeben sich die folgenden Lift-Werte (gerundet auf zwei Nachkommastellen):
   - $Lift_1 = 1 / 0{,}4 = 2{,}50$
   - $Lift_2 = 2 / 0{,}8 = 2{,}50$
   - $Lift_3 = 3 / 1{,}2 = 2{,}50$
   - $Lift_4 = 3 / 1{,}6 = 1{,}88$
   - $Lift_5 = 3 / 2{,}0 = 1{,}50$
   - $Lift_6 = 4 / 2{,}4 = 1{,}67$
   - $Lift_7 = 4 / 2{,}8 = 1{,}43$
   - $Lift_8 = 4 / 3{,}2 = 1{,}25$
   - $Lift_9 = 4 / 3{,}6 = 1{,}11$
   - $Lift_{10} = 4 / 4 = 1{,}00$

   Mit den Liftwerten ergibt sich folgender Lift-Chart



Für die Erstellung eines ROC-Charts müssen die True Positive Rate (TPR) und die False Positive Rate (FPR) für verschiedene Klassifikationsschwellwerte ermittelt werden. Hierbei gilt:

- TPR = Anteil der korrekt als positives Ereignis prognostizierten Fälle im Verhältnis zu allen positiven Ereignissen
- FPR = Anteil der fälschlicherweise als positives Ereignis prognostizierten Fälle im Verhältnis zu allen negativen Ereignissen

Im Folgenden sind die TPR und FPR für exemplarische Klassifikationsschwellwerte ermittelt.

		Prognose Totalschaden bei Cut-Off von …										
Totalschaden-prognose Modell 1	**Fahrzeugzustand**	100%	90%	80%	70%	60%	50%	40%	30%	20%	10%	0%
3%	Kein Totalschaden	0	0	0	0	0	0	0	0	0	0	1
38%	Kein Totalschaden	0	0	0	0	0	0	0	1	1	1	1
88%	Totalschaden	0	0	1	1	1	1	1	1	1	1	1
67%	Totalschaden	0	0	0	0	1	1	1	1	1	1	1
21%	Totalschaden	0	0	0	0	0	0	0	0	1	1	1
63%	Kein Totalschaden	0	0	0	0	1	1	1	1	1	1	1
7%	Kein Totalschaden	0	0	0	0	0	0	0	0	0	0	1
9%	Kein Totalschaden	0	0	0	0	0	0	0	0	0	0	1
83%	Totalschaden	0	0	1	1	1	1	1	1	1	1	1
5%	Kein Totalschaden	0	0	0	0	0	0	0	0	0	0	1
	TPR	**0%**	**0%**	**50%**	**50%**	**75%**	**75%**	**75%**	**75%**	**100%**	**100%**	**100%**
	FPR	**0%**	**0%**	**0%**	**0%**	**17%**	**17%**	**17%**	**33%**	**33%**	**33%**	**100%**

Aus den TPR und FPR ergeben sich die Datenpunkte im ROC-Chart.

c) Bagging (Bootstrap aggregating) ist eine Ensemble-Methodik zur Verbesserung der Prognosegüte eines Prognosemodells. Ziel und Vorteil des Bagging ist es, die Varianz von einem Prognosemodell zu reduzieren.

Zur Anwendung von Bagging wird ein Algorithmus zu einem Prognosemodell mehrfach angewendet. Für eine gegebene Datengrundlage werden mehrere zufällige Stichproben erzeugt. Hierbei werden die Daten mit Ersetzen (with replacement) zufällig aus der Datengrundlage gezogen. Mit den verschiedenen Stichproben werden verschiedene Prognosemodelle erzeugt. Aus den Prognosewerten aller Prognosemodelle erfolgt über eine Aggregation (z.B. Selektion der häufigsten Ausprägung zu den Prognosen bei Klassifikationsmodellen) die Vorhersage.

d) Die Erstellung des Random Forest mit der AdaBoost-Methode erfolgt in den nachfolgenden Schritten. Hierbei sollen N Entscheidungsbäume erzeugt werden.

- Schritt 0: Festlegung der initialen Gewichte w_i pro Datenpunkt i mit $w_i = \frac{1}{N_{ges}}$ (mit N_{ges} Datenpunkten in den Trainingsdaten).
- Schritt 1: Training des j-ten Entscheidungsbaums (mit $j \in N$ beginnend mit j = 1).
- Schritt 2: Ermittlung der gewichteten Fehlerrate e_j für den j-ten Entscheidungsbaum. Hierbei wird die Fehlerrate jedes Datenpunkt mit w_i gewichtet.
- Schritt 3: Ermittlung der Gewichtung des Entscheidungsbaums w_tree_j in dem Ensemble (Random Forest):
  - $w\_tree_j = \eta * \log\left(\frac{1-e_j}{e_j}\right)$
    - (hierbei Verwendung der Learning Rate η als Parameter im Modell.)
- Schritt 4: Update der Gewichte w_i pro Datenpunkt i
  - $w_i = \begin{cases} w_i & falls\ y_i\ korrekt\ vorhergesagt\ wurde \\ w_i * e^{w\_tree_j} & falls\ y_i\ nicht\ korrekt\ vorhergesagt\ wurde \end{cases}$
- Schritt 5: Wiederholung von Schritt 1 bis 4 bis die maximale Anzahl N von Entscheidungsbäumen erzeugt wurde.
- Schritt 6: Ermittlung der finalen Vorhersage in dem Ensemble-Modell (Random Forest) unter Berücksichtigung der Gewichtung w_tree_j zu jedem Entscheidungsbaum.

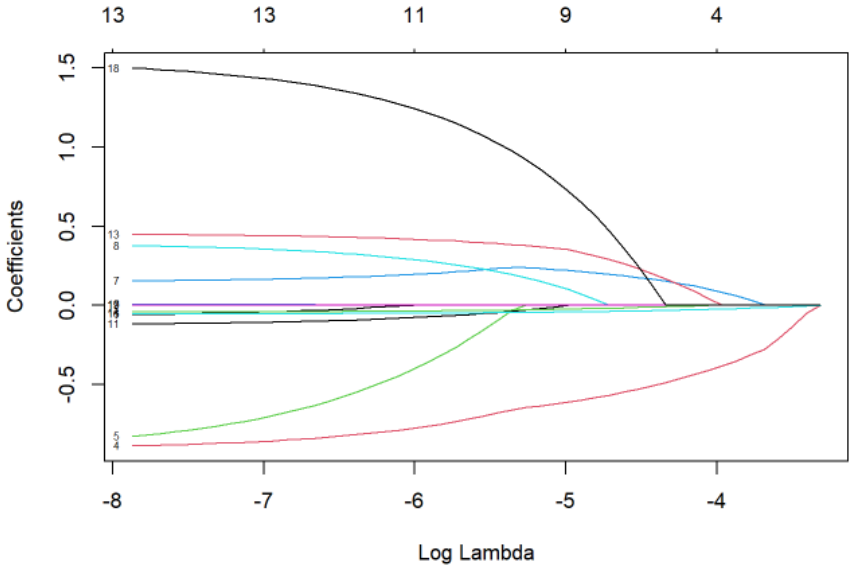
Aufgabe 3 [Modellselektion und Regularisierung 2 (7.4.2), Data Mining 2 (8.1.1, 8.1.7)] (38 Punkte)

Mit einem GLM führen Sie Regularisierungen mittels Ridge und Lasso durch.

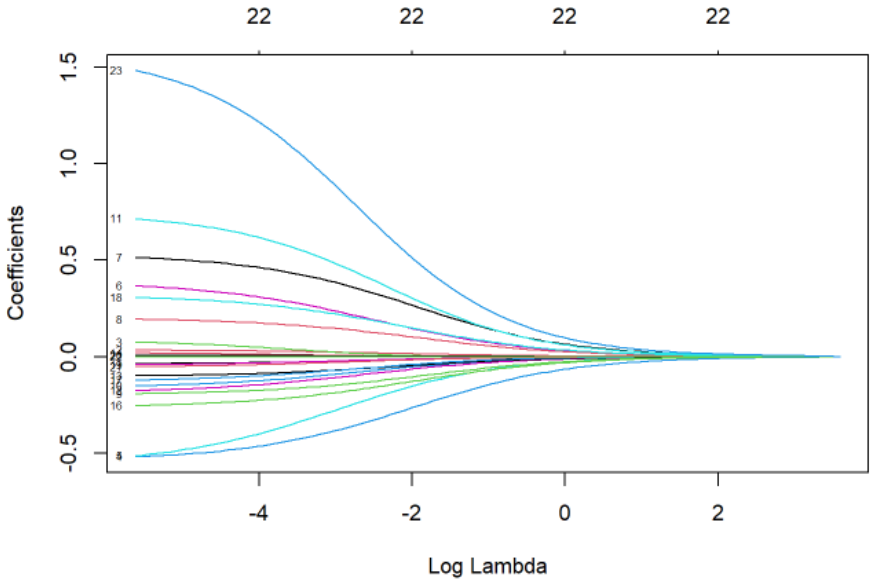
a) (8 Punkte) Erläutern Sie die beiden Verfahren und arbeiten dabei Gemeinsamkeiten und Unterschiede heraus.

Folgende Grafiken wurden beim Einsatz von Ridge und Lasso erstellt:

G1:

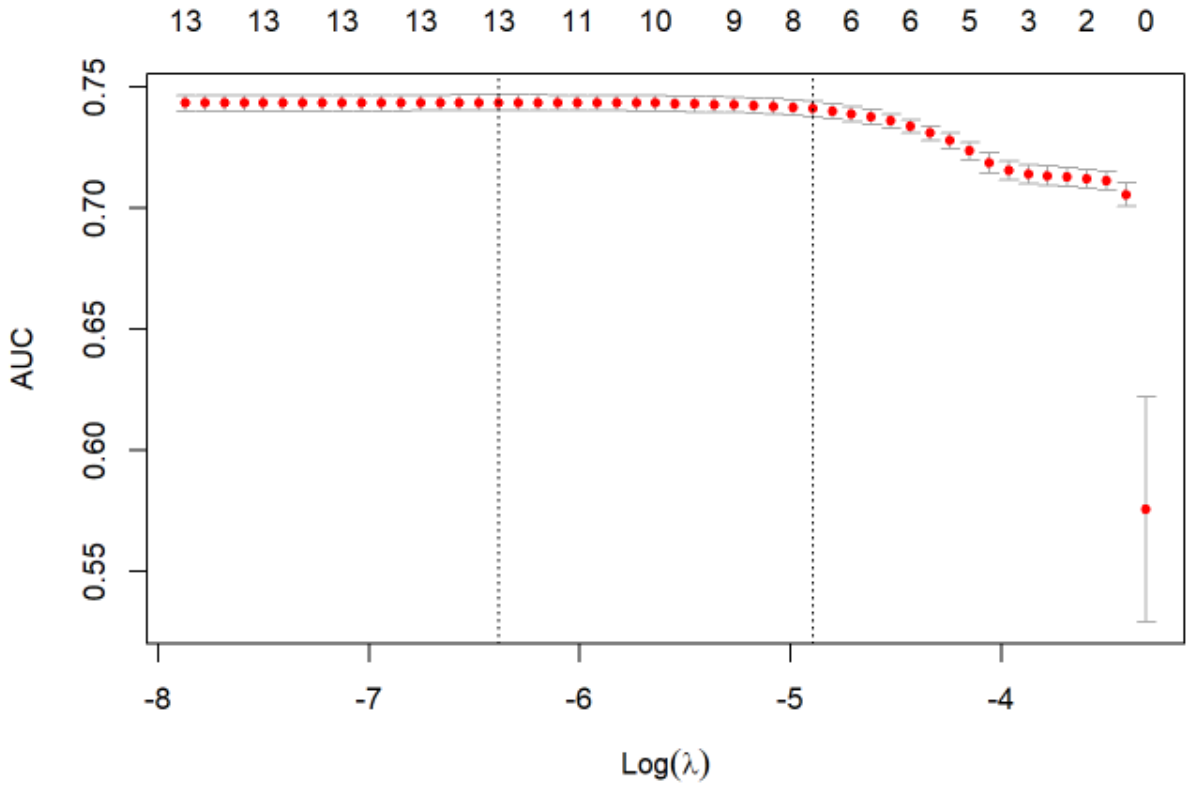


G2:



b) (7 Punkte) Beschreiben Sie im Detail, was die beiden Grafiken G1 und G2 zeigen. Erläutern Sie insbesondere die verschiedenen Beschriftungen. Wofür stehen die Kurven? Welche Grafik gehört zu Ridge und welche zu Lasso und warum?

Außerdem erhalten Sie die folgende Grafik G3.



c) (8 Punkte) Beschreiben Sie im Detail, was die Grafik G3 zeigt. Erläutern Sie insbesondere die verschiedenen Beschriftungen. Wofür stehen die Punkte und Linien und weshalb sind diese hier eingezeichnet? Gehört G3 zu G1 oder zu G2 und warum (nennen Sie mindestens zwei Gründe)?

Die Grafiken G1 bis G3 wurden mit folgendem R-Code erzeugt (wobei die Markierungen ①-⑤ nicht zum Code gehören, sondern lediglich für Sie als Referenzierungshilfe dienen sollen):

```
[…]

① mod1 <- cv.glmnet(Storniert ~ ., data=train, family="binomial",
  type.measure ="auc", nfolds=5, alpha=0, parallel=TRUE)

② plot(mod1$glmnet.fit, xvar = "lambda", label = TRUE)
```

③
```
mod2 = cv.glmnet(Storniert ~ . -Regio, data=train, family="bino-
mial", type.measure ="auc", nfolds=5, alpha=1, parallel=TRUE)
```

④
```
plot(mod2)
```

⑤
```
plot(mod2$glmnet.fit, xvar = "lambda", label = TRUE)
```

[…]

d) (10 Punkte) Was bedeutet der im Code zweimal verwendete Parameter alpha und welcher Modellansatz steckt dahinter? Erläutern Sie, welcher Funktionsaufruf (①-⑤) welcher Grafik G1 bis G3 zugrunde liegt und warum. Beschreiben Sie in ein bis zwei Sätzen die Bedeutung des im Code zweimal verwendeten Parameter nfolds.

e) (5 Punkte) Nennen Sie die Phasen des Prozessmodells CRISP-DM (durchgängig Deutsch oder Englisch). Welcher Phase und welcher Aufgabe würden Sie die Grafiken G1 bis G3 zuordnen und weshalb?

Lösungsvorschlag:

a) Ridge und Lasso sind zwei Verfahren zur **Regularisierung** von GLMs oder auch nicht-linearen Modellen, also Modellanpassungen zur **Senkung der Varianz** der Vorhersage. Genauer handelt es sich um **Shrinkage**-Verfahren, bei denen durch eine geeignete Loss-Funktion die geschätzten Modell-**Parameter möglichst niedrige Absolutwerte** annehmen.

Bei Ridge Regression wird die zu minimierende **Loss-Funktion** um einen **Strafterm** mit der **L2-Norm der Modell-Parameter β** ergänzt, also

$$Loss_{Basismodell}((x_{ij}),(y_i),(\beta_0,\dots,\beta_p)) + \boldsymbol{\lambda}\sum_{j=1}^{p}\beta_j^2$$

mit der L^2-Norm

$$||(\beta_1,\dots,\beta_p)||_2 = \boldsymbol{\lambda}\sum_{j=1}^{p}\beta_j^2.$$

Analog dazu wird bei der LASSO-Methode die Loss-Funktion um einen **Strafterm** mit der **L1-Norm der Modell-Parameter β** ergänzt

$$Loss_{Basismodell}((x_{ij}),(y_i),(\beta_0,\dots,\beta_p)) + \boldsymbol{\lambda}\sum_{j=1}^{p}|\beta_j|$$

mit der L^1-Norm

$$||(\beta_1,\dots,\beta_p)||_1 = \boldsymbol{\lambda}\sum_{j=1}^{p}|\beta_j|.$$

In beiden Verfahren

- führt die Minimierung der Loss-Funktion über β zu Schätzwerten β_1, …, β_p mit **verhältnismäßig kleinen Absolutwerten** und damit zu einer **Senkung der Varianz** und **Reduktion des Auftretens von Overfitting**.

- bestimmt der Regularisierungsparameter (Tuning-Parameter) $\lambda > 0$ den **Grad der Regularisierung**. Dabei bewirken **λ-Werte nahe 0**

praktisch keine Regularisierung (die Loss-Funktion ist wie beim Originalmodell), während **sehr große λ-Werte** (λ -> ∞) die Koeffizienten immer kleiner werden lassen.

Bei LASSO (least absolute shrinkage and selection operator) kommt es häufig vor, dass manche der geschätzten Modell-Parameter mit wachsendem λ nicht nur betragsmäßig kleiner werden, sondern sogar exakt den Wert 0 annehmen. Bei Ridge hingegen werden die Absolutwerte der geschätzten Modell-Parameter kleiner, bleiben in der Regel aber größer als Null. Mit anderen Worten: LASSO führt im Unterschied zu Ridge nicht nur zur Regularisierung, sondern auch zu einer **automatischen Merkmalsauswahl** (**Feature Selection**).

Diese Reduktion der Variablenanzahl erhöht die **Interpretierbarkeit des Modells** gegenüber einem Ridge-Modell mit voller Variablenanzahl.

b) G1 und G2 zeigen die Werte der Koeffizienten in Abhängigkeit von log(λ).

y-Achse: Werte der einzelnen Koeffizienten.

x-Achse: log(λ) mit λ = Regularisierungsparameter gemäß Teilaufgabe (a).

Die einzelnen Kurven gehören zu jeweils einem der im Modell enthaltenen Parameter β_j und verdeutlichen so die Regularisierung mit wachsendem λ.

Die Beschriftung am oberen Rand der Grafik zeigt in Abhängigkeit vom jeweiligen log(λ) die Anzahl der im Modell enthaltenen Modell-Parameter (Koeffizienten) ungleich 0.

In G1 und G2 nähern sich die Kurven grundsätzlich immer mehr der Nulllinie. Diese Annäherung ist bei G2 lediglich asymptotisch, während bei G1 der Wert Null tatsächlich erreicht wird.

Somit gehört G1 zu LASSO und G2 zu Ridge.

c) G3 zeigt einen Plot des Kreuzvalidierungsfehlers in Abhängigkeit von log(λ) bei Nutzung der AUC-Metrik.

y-Achse: AUC = Area Under Curve, wobei mit Curve die ROC-Kurve gemeint ist.

x-Achse: log(λ) mit λ = Regularisierungsparameter gemäß Teilaufgabe (a).

Die roten Punkte sind somit der bei der Kreuzvalidierung gefundene mittlere AUC zum jeweiligen log(λ) und die Error bars (Fehlerbalken), die die Range von +/- dem einfachen Standardfehler anzeigen.

Die Beschriftung am oberen Rand der Grafik zeigt in Abhängigkeit vom jeweiligen log(λ) die Anzahl der im Modell enthaltenen Modell-Parameter (Koeffizienten) ungleich 0.

G3 „gehört zu G1", sprich: gehört ebenso wie G1 zum LASSO-Modell. Dies sieht man nicht nur an den gleichen Werten auf der x-Achse (für log(λ)), sondern auch an den abnehmenden Werten oberhalb der Grafik, die bei G1 wie bei G3 die abnehmende Anzahl an Parametern anzeigen. Diese abnehmende Anzahl ist aber typisch für LASSO und kommt bei Ridge praktisch nicht vor (siehe G2).

Zu den beiden vertikalen gepunkteten Linien:

- Die linke (bei etwa log(λ) = -6,4) markiert das λ mit dem maximalen AUC-Wert, das sogenannte lambda.min.
- Die rechte (bei etwa log(λ) = -4,8) markiert das sogenannte 1-Standardfehler-λ, kurz: lambda.1se. Dieses erhält man, indem man das größte λ sucht, für das der zugehörige AUC-Wert noch innerhalb der zu lambda.min gehörigen Fehlerbalken liegt.

d) Der **Parameter alpha** liegt der Vereinheitlichung von Ridge und LASSO in Form des sog. Elastic Net zugrunde. In dem R package glmnet wird der Strafterm mit einem Mischungsparameter α definiert. Genauer: Der Strafterm ist hier

$$\lambda\left[(1-\alpha)\frac{1}{2}\sum_{j=1}^{p}\beta_j^2+\alpha\sum_{j=1}^{p}|\beta_j|\right]$$

Reines Lasso ist α = 1 (Funktionsaufruf ③), reines Ridge ist α = 0 (Funktionsaufruf ①).

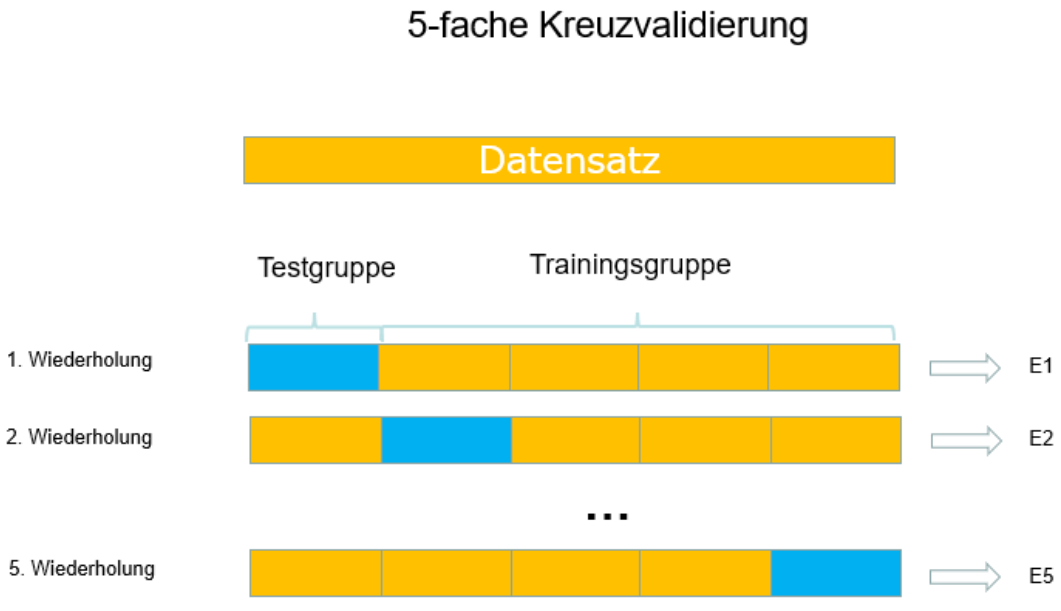
Die **Funktionsaufrufe** ②, ④ und ⑤ erzeugen Plots, also Grafiken. Aufgrund obiger Identifikation von Lasso und Ridge gehört ② zu Ridge, ④ und ⑤ zu Lasso.

Die Aufrufe ② und ⑤ erhalten jeweils das gefittete Elastic Net als Parameter, der Aufruf ④ dagegen das gesamte Modell. Da G1 und G2 jeweils den

gleichen Grafiktyp haben, G3 aber ein anderer Grafiktyp ist, muss also folgender Zusammenhang gelten:

Funktionsaufruf	Lasso / Ridge	Grafik
②	Ridge	G2
④	Lasso	G3
⑤	Lasso	G1

Der **Parameter nfolds** gibt an, eine wievielfache Kreuzvalidierung durchgeführt werden soll. Hier wird also jeweils eine 5-fache Kreuzvalidierung durchgeführt. Dabei wird der gesamte Datensatz zufällig in 5 gleich große Teile zerlegt und jeweils das Training auf nur 4/5 des Datensatzes durchgeführt, während das restliche Fünftel zur Validierung (in der folgenden Grafik als Testgruppe bezeichnet) herangezogen wird.



$$E = \frac{1}{5}\Sigma E_i$$

Anm.: Diese Grafik dient nur der Illustration und ist kein Bestandteil der geforderten Lösung.

e) Die 6 Phasen von CRISP-DM heißen

- Geschäftsverständnis (Business Understanding)
- Datenverständnis (Data Understanding)
- Datenvorbereitung (Data Preparation)
- Modellierung (Modelling)
- Evaluierung (Evaluation)
- Bereitstellung / Anwendung (Deployment)

Die Grafiken G1 bis G3 werden (neben anderen Werten und Grafiken) benutzt, um die Entscheidung für ein bestimmtes λ und damit für ein bestimmtes Modell zu treffen. Der Haupteinsatz ist damit in der Phase Modellierung im Schritt Bewerten des Modells.

Aufgabe 4 [5.1 Gesellschaftliches Umfeld & Ethik 2, 5.2 Datenschutz, 8.3. Innovative Produkte 2] (28 Punkte)

Sie lesen auf der Rückfahrt von der Jahrestagung der DAV im Zug ein Interview einer großen deutschen Wirtschaftszeitung mit einem Vorstandsvorsitzenden eines großen IT-Dienstleisters. Sie sind auf den Artikel aufmerksam geworden, da die Überschrift „KI und Chatbots bringen enorme Möglichkeiten für die Versicherer" lautet und Sie sich gerade in Ihrem Unternehmen mit diesen Themen beschäftigen.

a) [6 Punkte] Nennen Sie drei sinnvolle Einsatzfelder für Chatbots in Ihrem Unternehmen und die jeweiligen Chancen.

b) [6 Punkte] Nennen Sie drei Gründe, warum der Einsatz von Chatbots problematisch oder schwierig sein kann, und erläutern Sie diese.

Der Vorstand Ihres Versicherungsunternehmen hat entschieden, dass er Chatbots einsetzen möchte. Dabei möchte man gerne einen Chatbot eines amerikanischen Anbieters aus dem Silicon Valley benutzen. Der Chatbot beruht auf den sogenannten Transformer-Modellen. Der Service läuft auf einem US-amerikanischen Server und wird über eine kostenpflichtige API aufgerufen. Das zugrundliegende Modell hat mehrere hundert Milliarden Parameter und wurde mit über einem Terabyte Daten trainiert. Die Modellarchitektur ist bekannt, aber Sie haben keinen Zugriff auf die konkreten Modellparameter. Der Vorstand bittet Sie nun, mögliche weitere Risiken, die sich aus Nutzung dieses Anbieters, zu beurteilen.

Hinweis: Bei den Transformer-Modellen handelt es sich um eine spezielle Klasse von neuronalen Netzen. Für die Beantwortung werden aber keine Kenntnisse zu neuronalen Netzen oder Transformer vorausgesetzt, da diese erst in den Modulen Immersion und Completion behandelt werden.

c) [10 Punkte] Nennen Sie fünf mögliche Risiken, die sich aus der Nutzung dieses Anbieters ergeben, und erläutern Sie diese.

Da der Vorstand Sie beauftragt hat, sich in das Thema weiter einzuarbeiten, lesen Sie nun wissenschaftliche Artikel zu Transformer-Modellen. Dabei erfahren Sie u.a. folgendes:

Seit 2017 sind sogenannte Transformer-Modell wie etwa *Bert* und *GPT-3* in der Entwicklung. Diese Modelle werden auf sehr großen Datenmengen (Gigabytebereich) aus dem Internet wie z.B. Wikipedia, Zeitungsartikel oder Webseiten trainiert und sind in der Lage, natürliche Sprache zu verarbeiten. Diese Transformer-Modelle zeigen beindruckende Fähigkeiten. So sind sie beispielsweise in der Lage,

Texte zusammenzufassen, Fragen zu beantworten, Emotionsanalyse durchzuführen, zu chatten und Programmcode zu erstellen. Die neuesten Modelle können sogar Witze erklären.

Transformer-Modelle zeigen einige Schwachstellen. Sie reproduzieren beispielsweise Stereotypen über Geschlechterrollen, Religionen und ethnische Gruppen. Außerdem können sie auch Beleidigungen und unangemessene Sprache benutzen. Die Ursache dafür sind in erster Linie die Trainingsdaten. Sie enthalten ungefilterten Text aus dem Internet, in denen Stereotypen vorkommen. In den Trainingsdaten sind weiterhin z.B. auch Kontaktinformationen von Personen enthalten, sodass Transformer-Modelle auch datenschutzrechtlich relevante Daten reproduzieren können.

Die ökologischen und finanziellen Kosten von solchen Modellen sind auch sehr hoch. So verbraucht das Training eines aktuellen Transformer-Modells Energie in der Größenordnung eines Langstreckenfluges und kostet mehrere Millionen US-Dollar.

d) [6 Punkte] Der Vorstand bittet Sie nun, drei weitere Risiken, die sich aus Nutzung von Transformer-Modellen ergeben, zu beurteilen. Er bittet Sie, dass Sie dabei Ihr Wissen aus der CADS-Ausbildung über gesetzliche Vorgaben und Reputationsrisiken einbringen.

Lösungsvorschlag:

a)

(i) Kundenbetreuung: Chance ist Kundenbetreuung zu jeder Zeit und ohne Beschränkung durch nicht genügend Personal.

(ii) Automatisierte Schadenabwicklung: Chance ist eine schnellere Abwicklung, da der Chatbot zu jedem Zeitpunkt kontaktiert werden kann und dieser unmittelbar eine Rückmeldung geben kann.

(iii) Vermittlung und Beratung von Kunden: Es können Kunden gewonnen werden, die sich lieber online beraten lassen.

b)

(i) Sonderfälle bzw. besondere Sachverhalte können nicht durch Chatbots bearbeitet werden.

(ii) Kundenzufriedenheit: Einige Kunden bevorzugen lieber eine Kommunikation mit einem echten Kundenbetreuer.

(iii) Manipulationsmöglichkeit: Sollte der Chatbot nicht nur allgemeine Informationen, sondern auch kundenspezifische Anfragen beantworten können, so besteht z.B. ein Datenschutzrisiko.

c)

(i) Die Übermittlung von Daten, die der DSVGO unterliegen, in die USA ist rechtlich problematisch.

(ii) Das Modell ist eine Blackbox. Das Modell wurde nicht selbst erstellt und man hat auch keinen direkten Zugriff auf das Modell. Es ist daher nicht möglich, das Modell selbst zu überprüfen.

(iii) Bei mehreren hundert Milliarden Parametern ist das Modell als solches auch schwierig zu überprüfen. Die Interpretierbar und Validierbarkeit des Modells ist damit nicht gegeben und das Risiko von Fehlentscheidungen des Modells ist daher hoch.

(iv) Es besteht ein Kostenrisiko, da der Anbieter die Gebühren im Laufe der Zeit erhöhen kann.

(v) Der Chatbot ist nicht auf den speziellen deutschen Versicherungsmarkt zugeschnitten. Das kann dazu führen, dass die Qualität des

Chatbots nicht ausreichend ist. Das kann beispielsweise zu nicht sachgerechten Antworten und zu Kundenunzufriedenheit führen.

d)

(i) Datenschutz: Es dürfen keine persönlichen Daten ohne Einwilligung gespeichert werden. Wenn ein Modell solche Daten reproduziert, stellt es eine Datenschutzverletzung dar, da die Daten im Modell gespeichert sind und die Daten ohne Einwilligung verarbeitet werden. Es gibt auch nicht die Möglichkeit, einzelne Daten aus dem Modell zu löschen.

(ii) Diskriminierung: Es darf keine Benachteiligung beispielsweise aufgrund der ethnischen Herkunft oder des Geschlechts geben.

(iii) Reputationsrisiko: Der Einsatz von Modellen, die sehr viel Ressourcen verbrauchen und zu Diskriminierung und Datenschutzverletzung neigen, stellen auch ein hohes Reputationsrisiko dar. Außerdem kann der Einsatz solcher großen Modelle, die nur schlecht überprüfbar sind, einen Vertrauensverlust darstellen. Ein Modell, dass zu Beleidigungen neigt, kann nicht für den Kundenkontakt benutzt werden.

Aufgabe 5 [6.2 Datenverarbeitungstechnologien 2] (40 Punkte)

Der Ausschließlichkeit-Vertrieb hat mit einem Online-Tool die Möglichkeit, nach einem Beratungsgespräch die Einschätzung der Personen zum Gespräch einzuholen. Dabei bekommen die Personen die Möglichkeit, eine Sternenbewertung (Beste Bewertung: 5 Sterne, schlechteste Bewertung: 1 Stern) sowie einen Kommentar abzugeben. Insgesamt liegen mehrere Millionen Bewertungen vor. Exemplarisch werden einige Datensätze angegeben:

Nummer	**Kommentar**	**Bewertung**
100.581	Unkompliziert, bedarfsgerecht und kundenorientiert! So muss Beratung sein!	5
306.231	Unterirdischer Kundenservice	1
1.451.321	Trotz Rücktritt vom Vertrag kam die Versicherungspolice. Auf Fragen zur Versicherung keine Antwort!	1
1.895.324	Schnell und zuverlässig, beigefügte Unterlagen könnten konkreter und umfangreicher sein.	4
2.010.491	Kann ich nur empfehlen!	4
2.391.431	Ich bin begeistert!	5

Ziel ist es, zu ermitteln wie viele Bewertungen in die Kategorien 1-5 fallen.

a) [9 Punkte] Grenzen Sie die Imperative Programmierung gegen die Funktionale Programmierung ab, indem Sie mindestens drei Merkmale für jede Seite nennen.

b) [9 Punkte] Erklären Sie die Funktionale `Map`, `Filter` und `Reduce`. Geben Sie hierfür jeweils die mathematische Notation, ein Beispiel in Pseudocode, sowie eine ergänzende Erklärung in kurzen Sätzen an.

c) [15 Punkte] Geben Sie für die Beispieldatensätze der Kommentare und Bewertungen oben einen algorithmischen Grundablauf an, wie mit Hilfe von

`Map`, `Filter` und `Reduce` ermittelt werden kann, wie viele Bewertungen in welche Kategorie fallen. Nehmen Sie dabei an, dass die Daten bereits auf dem Cluster im HDFS-Format gespeichert und daher bereits partitioniert sind. Die Partitionen sind dabei so eingeteilt, dass alle Daten mit Nummer kleiner gleich 1.000.000 in Partition 1 liegen, alle Daten mit Nummern > 1.000.000 und <= 2.000.000 in Partition 2 etc. Berücksichtigen Sie zudem eine Anzahl von R=2 Prozessen. Der Ablauf ist auch visuell darzustellen.

d) [7 Punkte] Beschreiben Sie die Job-Steuerung in Hadoop. Gehen Sie dabei insbesondere auf die Elemente Master Node, Slave Node, Job Tracker und Task Tracker ein, welche in der vereinfachten Job-Steuerung in Hadoop-Version 1.X verwendet werden.

Lösungsvorschlag:

a) Funktionale Programmierung:

- Daten werden durch sukzessive Anwendung von Funktionen verarbeitet. Die Lösung eines Problems wird damit als Satz von ausführbaren Funktionen formuliert (deklaratives Programmierparadigma).
- Auf innere Zustände eines Berechnungsprozesses wird verzichtet, ebenso auf zugehörige Zustandsänderungen (Seiteneffekte).
- Funktionsdefinitionen können insbesondere bei Funktionsanwendungen ohne explizite Namensgebung literal in der Stellung des Funktionssymbols stehen („Lambdas").
- Andere Antwortmöglichkeiten sind möglich.

Imperative Programmierung:

- Ältestes Programmierparadigma. Ein Programm besteht aus einer klar definierten Abfolge von Handlungsanweisungen an einen Computer.
- Nutzung von Kontrollstrukturen wie Schleifen oder Verzweigungen.
- Probleme können bei der Verarbeitung großer Datenmengen auftreten, da es hierbei erforderlich ist, die Rechnungen auf vielen Kernen oder Prozessoren zu verteilen.
- Andere Antwortmöglichkeiten sind möglich.

b) Die folgenden Ausführungen orientieren sich an Python.

Map: Das **Map Funktional** wendet eine Funtion auf jedes Element einer Liste an und gibt eine Liste der gleichen Länge, gefüllt mit den Funktionswerten, zurück. Schreibweise:

map(funktion, iterable)

Mathematisch: $map\colon [x_1, x_2, \dots, x_n], f(x) \rightarrow [f(x_1), f(x_2), \dots, f(x_n)]$

Beispiel: Quadrieren der Zahlen einer List:

*squares = list(map(lambda x: x*x, [1,2,3,4,5]))*

Filter: Analog Map filtert dieses Funktional basierend auf einem iterierbaren Objekt die Elemente abhängig von einer vorgegebenen Funktion. Diese Funktion gibt als Rückgabewerte nur True/False zurück. Schreibweise:

filter(funktion, iterable)

Mathematisch: $filter\colon [x_1, x_2, \dots, x_n), c(x) \rightarrow [x_k, \dots, x_l), 1 \leq k < l \leq n$ für die eine Bedingung $c : R \rightarrow \{True, False\}$ erfüllt ist: $c(x_k) = \cdots = c(x_l) = True$.

Beispiel: Filtere Werte größer als 90:

highvalues = list(filter(lambda x: x > 90, [80,95,93,20]))

Reduce: Reduziert den übergebenen Input auf ein einziges Element. Schreibweise:

reduce(funktion, iterable)

Mathematisch: $reduce\colon [x_1, x_2, x_3, x_4], f(x, y) \rightarrow f(f(f(x_1, x_2), x_3), x_4)$

Die Funktion f ist assoziativ, und das Beispiel wird hier mit vier Elementen dargestellt.

Beispiel:

sumres = reduce(lambda a,b : a + b, [1,2,3,4])

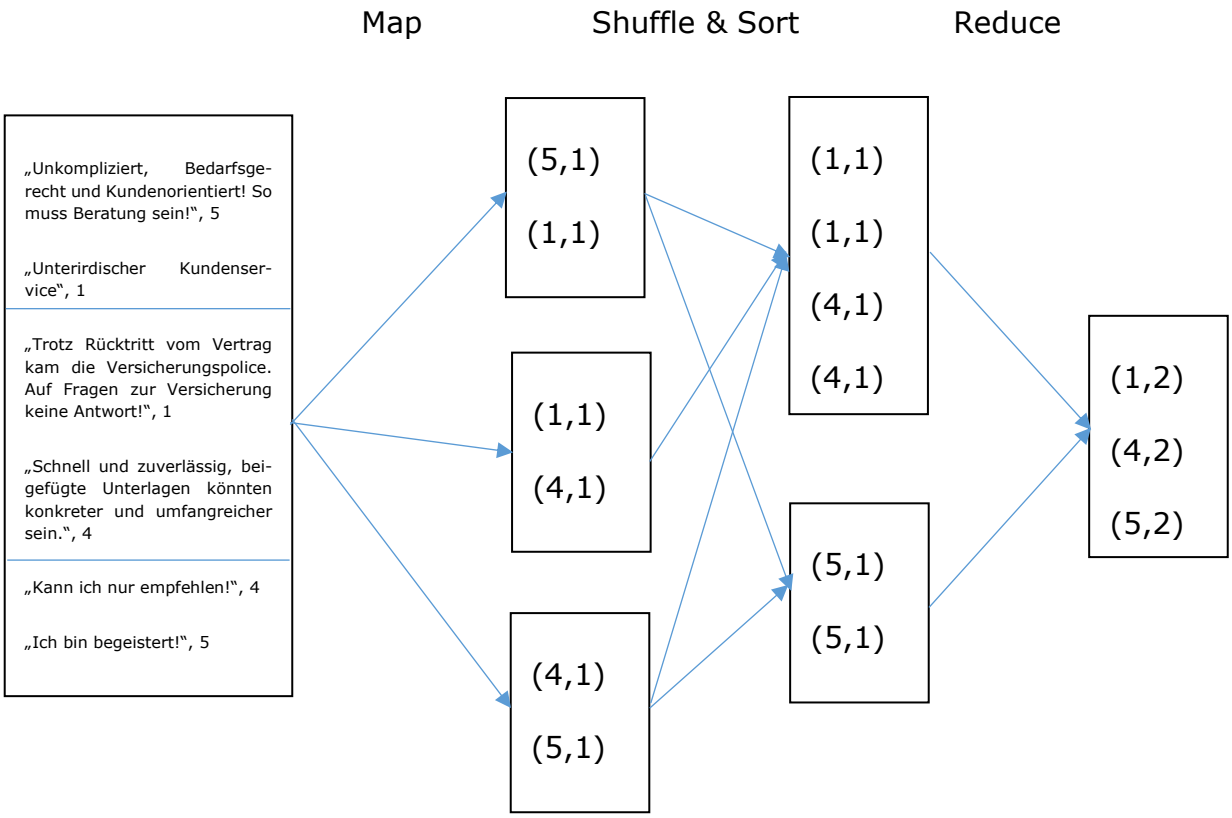
c) Nach Annahme befinden sich die ersten beiden Einträge der Tabelle in Partition 1, die nächsten beiden in Partition 2 sowie die letzten beiden in Partition 3. Ein Splitting ist daher nicht nötig. Für die Bearbeitung ist es notwendig, die Bewertung als Key zu berücksichtigen. Die Kommentare werden für die Auswertung nicht benötigt und werden daher durch eine „1" ersetzt.

Der Schritt Map sorgt dafür, dass die Bewertung als Key zur Verfügung steht. In diesem Schritt werden auch die Kommentare ersetzt.

Der Schritt „Shuffle and Sort" verteilt die Ergebnisse des Map Taks auf die R=2 Prozesse.

Im letzten Reduce-Schritt werden die Bewertungen final verdichtet.

Diese Schritte werden in folgendem algorithmischen Grundablauf visualisiert:



d) Der algorithmische Grundablauf aus Teil c) muss von einem Cluster organisiert werden. Für diese Steuerung sind auf dem Cluster verschiedene Software-Komponenten zuständig, u.a.

Master Node: Hier befinden sich zwei wesentliche Software-Komponenten: Der Name Node (verwaltet die Meta Daten des HDFS Filesystems) und der Job Tracker (orchestriert den Map/Reduce Ablauf auf dem Cluster).

Slave Nodes: Hier sind die Daten in Form von HDFS Blöcken gespeichert. Außerdem finden hier die Map und Reduce-Berechnungen statt, was auf dem jeweiligen Slave Node durch den Task Tracker verwaltet und überwacht wird.

Job Tracker: Empfängt vom Client Anfragen zur Ausführung von Map/Reduce Programmen, kommuniziert mit dem Name Node um den Speicherort der Daten zu bestimmen und findet für jede Aufgabe den optimalen Slave Node, basierend auf der Lage der Daten und der Verfügbarkeit von Rechenkapazität. Zudem überwacht er die einzelnen Task Tracker und übermittelt den Gesamtstatus zurück an den Client.

Task Tracker: Empfängt Map- und Reduce- Aufgaben vom Job Tracker. Zudem sendet er regelmäßig Statusinformationen über die laufenden Tasks an den Job Tracker.