**Projektbericht zum Modul Information Retrival und Visualisierung**

Sommersemester 2021

**Visualisierung von verschiedenen Weindaten**

Marie-Luise Lindner

Matrikelnummer: 211231747

GitHub Repository: <https://github.com/RicBre/Elm-Projekt-WineInformation>

# Inhaltsverzeichnis

[Inhaltsverzeichnis II](#_Toc82161120)

[1. Einleitung 1](#_Toc82161121)

[1.1 Anwendungshintergrund 1](#_Toc82161122)

[1.2 Zielgruppen 3](#_Toc82161123)

[1.3 Überblick und Beiträge 4](#_Toc82161124)

[2. Daten 5](#_Toc82161125)

[2.1 Technische Breitstellung der Daten 6](#_Toc82161126)

[2.2 Datenvorverarbeitung 6](#_Toc82161127)

[3. Visualisierung 7](#_Toc82161128)

[3.1 Analyse der Anwendungsaufgaben 7](#_Toc82161129)

[3.2 Anforderungen an die Visualisierungen 8](#_Toc82161130)

[3.3 Präsentation der Visualisierung 9](#_Toc82161131)

[3.3.1 Visualisierung Eins 9](#_Toc82161132)

[3.3.2 Visualisierung Zwei 10](#_Toc82161133)

[3.3.3 Visualisierung Drei 12](#_Toc82161134)

[3.4 Interaktion 12](#_Toc82161135)

[4. Implementierung 13](#_Toc82161136)

[5. Anwendungsfälle 14](#_Toc82161137)

[5.1 Anwendung Visualisierung Eins 14](#_Toc82161138)

[5.2 Anwendung Visualisierung Zwei 14](#_Toc82161139)

[5.3 Anwendung Visualisierung Drei 16](#_Toc82161140)

[6. Verwandte Arbeiten 17](#_Toc82161141)

[7. Zusammenfassung und Ausblick 18](#_Toc82161142)

[8. Literatur i](#_Toc82161143)

[Anhang ii](#_Toc82161144)

[Git Historie ii](#_Toc82161145)

# Einleitung

Bereits zur Zeit des antiken Griechenlands herrschte der Wunsch nach einer Maschine mit der Fähigkeit menschlichen Denkens. Mehr als einhundert Jahre bevor programmierbare Computer wissenschaftliche Wirksamkeit erzielten, stellte sich die Frage nach künstlicher Intelligenz (KI). [12] Heute ist diese fester Bestandteil des menschlichen Alltags. Es ist ein blühendes Feld mit vielen praktischen Anwendungen und aktiven Forschungsthemen. Im bestehenden gesellschaftlichen Kontext bildet maschinelles Lernen die Basis für automatisierte Routinearbeiten, das Verständnis von Sprache oder Bildern, Diagnosen in der Medizin, sowie der Unterstützung wissenschaftlicher Grundlagenforschung. [8] Sie sind geeignet zur Vorhersage von Ereignissen und Modellierung komplexer zeitabhängiger Systeme. Insbesondere finden sie Anwendung, wenn die Einflussfaktoren für einen bestimmten Ausgang nicht oder nur unvollständig bekannt und die Zusammenhänge komplex und nichtlinear sind. [16]

Automatisierte Routinearbeiten sind unter anderem im Sektor des Automobilvertriebes beobachtbar. Berichten prognostizieren ein Wachstum bei Neuwagen in den kommenden 5 Jahren durchschnittlich um 3,5 %. Ein Anstieg im Gebrauchtwagensektor sei von 5% zu registrieren. Kausal der gesteigerten Nachfrage des Gebrauchtwagenmarktes sind veränderte Bedürfnisstrukturen der Verbraucher zu verzeichnen. Primäres Ziel ist die Optimierung des Wiederverkaufspreises das Automobiles. Die Validierung dessen ist verbunden mit Werkstattbesuchen, sowie Kostenvoranschlägen und wird als zeit -und ressourcenaufwändiger Prozess betrachtet. Ein indisches Unternehmen namens „avl“ fokussiert diese Thematik in einem Projekt. Ziel ist die Entwicklung eines Modells für Drittunternehmen, zur Schätzung des Preises des Autos des Kunden im Online-Portal. Folglich erfolge eine Rationalisierung des Verkaufsprozesses von Gebrauchtwagen. Methodisch soll die Umsetzung dieses Projektes mit Hilfe von maschinellem Lernen auf Basis eines Trainingsdatensatzes von rund 7000 Daten über Gebrauchtwagen erfolgen. [10]

Der vorliegende Bericht nimmt Rückschlüsse auf diesen Sachverhalt und die vorliegenden Daten. Ziel ist die Aufarbeitung des Datensatzes, sowie dessen anschließende Visualisierung in drei differenzierten Visualisierungstechniken.

## Anwendungshintergrund

Um ein Verständnis für die Wichtigkeit einer signifikanten Datengrundlage für künstliche Intelligenzen zu schaffen, steht im Folgenden der Anwendungshintergrund im Fokus.

Die Anfänge der Entwicklung von Künstlicher Intelligenz fokussierten die Umsetzung von Problemen, die sich durch eine Liste formaler, mathematischer Regeln beschreiben lassen. Als dessen größte Herausforderung ergaben sich Fragen, welche der Mensch intuitiv und automatisiert löst, wie beispielhaft das Erkennen von Gesichtern in Bildern. Ansätze zur Lösung sind künstliche neuronale Netze oder Entscheidungsbäume. Diese simulieren Prozesse des Zentralnervensystems höherer Lebewesen. Im Detail bilden sie ein abstrahiertes Modell miteinander verbundener Neuronen ab, welche auf Grundlage spezieller Anordnung und Verbindung eine Klassifizierung unstrukturierter Daten ermöglichen. [1]

Der Aufbau eines künstlichen neuronalen Netzes definiert sich über ein Schichtsystem. In erster Instanz steht die Eingabeschicht aus Neuronen (Input Layer), welche mittels Zwischenneuronen über mehrere Schichten (Hidden Layer) verknüpft ist. Das Konstrukt endet mit einer Ausgabeschicht (Output Layer), welche Ergebnisse zurückliefert. [1] Abbildung 1 verweist auf diese Struktur.

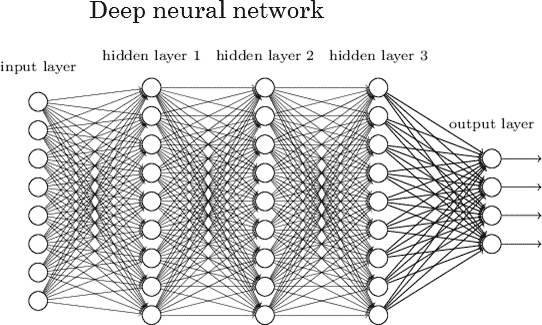


Abbildung 1: Struktur eines künstlichen neuronalen Netzes, Quelle: [3]

Der signifikante Unterschied zu klassischen statistischen Berechnungsverfahren definiert sich über die Lernfähigkeit eines künstlichen neuronalen Netzwerkes. Im Detail enthält ein solches System anfänglich keine Informationen.[16] Anhand von Trainingsdaten lernt das künstliche neuronale Netz. In diesem Prozess werden die Verknüpfungen, sowie die Verbindungsstärke innerhalb des Neuronenverbandes angepasst, sodass aus unbekannten Daten, Muster extrahiert und klassifiziert werden können. Die grundlegende Trainingsmethode erfordert große Datenmengen, welche den Analyseprozess stetig erneut durchlaufen. In jedem Analyseschritt gewonnene Erkenntnisse lassen sich mit weiteren Daten verknüpfen. Durch getroffene Entscheidungen erhalten die Verbindungen zwischen den Neuronen Gewichtungen. Positiv konjungierte Entscheidungen erhöhen das Gewicht der Verbindung. Ein Widerruf verringert das Gewicht. Folglich besitzt das künstliche neuronale Netzwerk die Fähigkeit Entscheidungen auf Basis von Verknüpfungen zu treffen.[1]

Auf Grundlage dessen wird die Bedeutung der Verfügbarkeit qualifizierter Daten- als Trainingsdatensatz- deutlich. Unzureichend verzerrte oder zu geringe Datenmengen führen zu verzerrten Ergebnissen, welche die Anwendung eines künstlichen neuronalen Netzes entkräften würden.[1] Entsprechend ist eine Analyse des Trainingsdatensatzes unabdinglich und von hoher Priorität. Aus diesem Grund wird im vorliegenden Bericht auf diese abgezielt.

Methodisch werden drei Visualisierungstechniken eingesetzt. Hauptziel ist es, anhand dieser den bereitgestellten Datensatz graphisch darzustellen und darauf basierend Rückschlüsse auf die Validität des Trainingsdatensatzes vornehmen zu können. Die graphische Umsetzung der Zusammenhänge gegebener Informationen ist vorteilhaft bei der Analyse von Ausreißern, der Betrachtung von Verhältnismäßigkeiten, sowie dem Erkennen signifikanter Werte eines Datensatzes. Auf Grundlage dessen wurden die Visualisierungstechniken der Baumhierarchie, des Scatterplots und der parallelen Koordinaten ausgewählt. Diese besitzen zu Teilen die Disposition der Interaktivität. Diese Eigenschaft ermöglicht den Nutzer den Vergleich von mehr als 2 Charakteristika der Informationen.

Die Visualisierungstechniken werden nachfolgend kurz vorgestellt.

Die erste Technik kennzeichnet eine Baumhierarchie. Die Baumhierarchie ist ein abstrakter Datentyp, welcher es ermöglicht hierarchische Strukturen abzubilden. Ausgehend von einem Wurzelelement können mehrere gleichartige Objekte miteinander verkettet werden, sodass lineare Strukturen aufgebrochen werden und folglich eine Verzweigung stattfindet. Diese Verzweigung kann beliebig oft wiederholt werden, welches das Addieren von Hierarchieebenen beinhaltet. Entsprechend ist Baumhierarchie geeignet Daten zu ordnen und hierarchische Zusammenhänge dazustellen. [9]

Bei der zweiten Visualisierungstechnik handelt es sich um einen Scatterplot. Mit wird die Gegenüberstellung zweier verschiedener numerischer Variablen ermöglicht. Die Werte werden in einem kartesischen Koordinatensystem eingetragen. Dies bietet die Möglichkeit die Beziehung zwischen diesen Variablen zu beobachten. Korrelationsbeziehungen, Muster oder Trends können erkannt und analysiert werden. Additional können mit Hilfe dieser Technik Daten gruppiert werden, welches Rückschlüsse auf mögliche Ausreißer oder Datenlücken vereinfacht. [17]

Die dritte Visualisierungstechnik repräsentiert den Ansatz der Parallelen Koordinaten. Auf Grund ihrer Beschaffenheit ergibt sich der Vorteil der Untersuchung mehrdimensionale Daten im zweidimensionalen Raum auf Trends, sowie Anomalien. Im Gegensatz zum Scatterplot, in welchem die zwei Koordinatenachsen rechtwinklig zueinander angeordnet sind, verlaufen die Achsen der Parallelen Koordinaten parallel und in gleichem Abstand. „Jede Linie von links nach rechts entspricht einem Datenpunkt und wird durch einen Polygonzug mit Ecken auf den parallelen Achsen dargestellt. Die Position der Ecke auf der i-ten Achse entspricht der i-ten Koordinate des Punktes.“ [2]

## Zielgruppen

Die Zielgruppen der graphischen Umsetzung des Trainingsdatensatzes lassen sich anhand des Daten Lebenszyklus herauskristallisieren. Dieser beinhaltet fünf Schritt. Schritt ein wird als „Data Collection“ (deutsch: Daten Kollektion) definiert. Dieser erläutert das Zusammentragen von Daten. Nachdem die Daten aufgenommen wurden, müssen sie in Formaten und an Orten gespeichert werden, an denen sie für andere Dienste leicht zugänglich sind. Dieser Vorgang beschreibt das „Data Cleaning“ (deutsch: Daten Säuberung). Um aus den erfassten Daten Erkenntnisse und Informationen zu gewinnen, müssen diese im dritten Schritt „Exploratory Data Analysis“ (deutsch: Explorative Daten Analyse) verarbeitet und analysiert werden. Im Detail bedeutet dies die Normalisierung aus der Quelle, Bereinigung und Speicherung in Analysesystemen, welche Abfragen und Untersuchungen ermöglichen. Schritt vier beschreibt das „Model Building“ (deutsch: Model Bauen). In diesem Vorgang werden Rückschlüsse aus der Analyse gezogen. Im finalen Schritt dem „Model Deployment“ (deutsch: Modell-Einsatz) wird die Bereitstellung von Modellen mit einer Daten-Pipeline in einer Unternehmensumgebung für die endgültige Benutzerabnahme angestrebt. Die Profession eines Data Scientist übernimmt die Aufgaben des gesamten Lebenszyklus. Der Fokus eines Data Engineers liegt vermehrt im Aufgabensektor der „Data Collection“ und des „Data Cleaning“. Die Rolle des Data Analyst zielt auf die Anforderungen des „Data Cleaning“ und der „Exploratory Data Analysis“ ab. Machine Learning Engineers verantworten die „Model Building“ und „Model Deployment“ Prozesse.[15] Da in diesem Bericht die Visualisierung des Trainingsdatensatzes, sowie dessen Analyse bearbeitet wird, kann in Konklusion von der Zielgruppe der Berufsbilder des Data Scientist, Data Analyst und des Maschine Learning Engineers ausgegangen werden. Berufsbedingt ist bei dieser Zielgruppe von einem exzellenten Vorwissen, sowie von Sicherheit im Umgang und der Auswertung der drei Visualisierungstechniken auszugehen. In diesem Zusammenhang ist eine Anwendung der gewonnen Kenntnisse im Rahmen einer Implementierung einer Maschine-Learning Anwendung in der Unternehmensumgebung denkbar. Additional besteht die Möglichkeit der Fehlererkennung im Datensatz und folglich dessen Ergänzung, Umstrukturierung oder Weiterverarbeitung.

Als zweite mögliche Zielgruppe können Studenten der Studiengänge im informatischen Sektor, wie beispielsweise Data Science, Informatik oder Automatisierungstechnik, betrachtet werden. Das deutsche Bildungssystem im universitären Bereich mit informatischer Ausprägung legt eine hohe Gewichtung auf die Ausbildung in mathematisch / statistischen Zusammenhängen, sowie analytischen Denken. Basierend dieser Annahme kann bei der Zielgruppe der Studenten von sehr guten Vorkenntnissen ausgegangen werden. Die Techniken des Baumdiagrammes, der parallelen Daten und des Scatterplots, sowie deren Auswertung gehören zur Grundausbildung eines Studenten mit wissenschaftlicher Ausrichtung. Eine mögliche Verwendung der Visualisierungen in diesem Kontext ist die des Lernobjektes. Die Visualisierungen ermöglichen einen fachlichen Zugewinn im Erkennen von Fehlern, Trends, Zusammenhängen oder Anomalien innerhalb des Datensatzes. Additional ist ein Einsatz als Grundlage in Projekten zum maschinellen Lernen im studentischen Kontext möglich.

Die dritte Zielgruppe wird durch Interessierte repräsentiert. Diese beschreibt Menschen, welche ein Grundinteresse für Gebrauchtwagen in Indien aufweisen. Da es sich in diesem Kontext vermehrt um eine Wissensaneignung auf persönlicher Ebene und weniger im beruflichen oder universitären Spektrum handelt, kann kein fachliches Grundwissen vorausgesetzt werden. Entsprechend besteht die Möglichkeit von geringen bis mäßigen Vorkenntnissen. Insbesondere im Hinblick auf die Techniken ist bei dieser Zielgruppe nicht davon auszugehen, dass die Funktionsweise der Parallelen Koordinaten bekannt ist. Trotzdem wird dieser Gruppe ausreichend Interesse unterstellt, Zeit und Energie in das Verständnis der Visualisierungstechnik zu investieren. Schlussfolgernd kann auch diese Visualisierungstechnik als Anwendung für interessierte Personen verstanden werden. In diesem Zusammenhang ermöglicht das Diagramm der Parallelen Koordinate eine Entnahme des Preis-Leistungs-Verhältnisses der einzelnen Objekte im Vergleich. Eine weitere relevante Visualisierungstechnik für diese Zielgruppe ist die explizite Baumhierarchie. Mit Hilfe dessen besteht die Möglichkeit einen Überblick über die Verfügbarkeiten von Gebrauchtwagenmodellen zu erhalten. Dadurch können Interessierte bei der Entscheidung eines Modelles unterstützt werden. Auf Basis der Einfachheit des Baumdiagrammes kann ein Verständnis der Funktionsweise dessen angenommen werden.

## Überblick und Beiträge

Die Informationsgrundlage dieses Projektes bilden Daten der Webseite „Kaggle“. [10] Dieser sind zwei Datensätze zu entnehmen. Ein Testdatensatz und ein Trainingsdatensatz. Dementsprechend ist davon auszugehen, dass diese im Zusammenhang mit Maschinellen Lernen stehen. Die Datensätze fokussieren Informationen über Gebrauchtwagen in Indien. Im Detail zentriert sich der Inhalt dessen auf die Komponenten *Name* (deutsch: Name), *Location* (deutsch: Ort), *Year* (deutsch: Jahr), *Kilometers\_Driven* (deutsch: gefahrene Kilometer), *Fuel\_Type* (deutsch: Kraftstofftyp), *Transmission* (deutsch: Schaltgetriebe), *Owner Type* (deutsch: Vorbesitzer), *Mileage* (deutsch: Kilometerstand), *Engine* (deutsch: Hubraum), *Power* (deutsch: Pferdestärken), *Seats* (deutsch: Sitze), *New\_Price* (deutsch: neuer Preis) und *Price* (deutsch: Preis). Erkennbar geht die Thematik des Gebrauchtwagenhandels mit einer Vielzahl an Informationen einher. Insbesondere in einem Verkaufsszenario, stellt ein hoher Informationsgehalt eine unabdingbare Notwendigkeit dar. Anhand genannter Eigenschaften besteht die Möglichkeit der Validierung des Preises eines gebrauchten Fahrzeugs. Handelsüblich wird in diesem Kontext auf die Expertise einer Fachkraft aus dem Autosektor zurückgegriffen. Werkstätten und Gebrauchtwagenhändler dienen als Bezugspunkte. Hingegen gilt der Besuch in einer Werkstatt als ressourcenaufwendig. Neben der energetischen Ressource stellt die zeitliche Komponente ein Problem dar. Aus diesem Grund besteht der Bedarf der Optimierung des Validationsprozesses von Gebrauchtwagen. Ein Online-Portal soll dem Nutzer die Möglichkeit geben durch Eingabe spezieller Kennzahlen seines Gebrauchtwagens, dessen Preis ermitteln zu können. In diesem Szenario entfällt der ressourcenaufwendige Prozess beim Fachmann. Um eine Ausgabe gültiger Werte der Software des Online-Portals sicher stellen zu können, ist der in Kapitel 1.1 beschriebene Prozess des maschinellen Lernens notwendig. Basierend eingespeister Trainingsdaten lernt das System und ist folglich fähig, anhand gegebenen Wertes, den Preis eines Gebrauchtwagens zu prognostizieren. Basierend dieser Grundlage ist die Unverzerrtheit des Trainingsdatensatzes unabdingbar. Schlussfolgernd wird eine genaue Analyse dessen als essentiell betrachtet. Aus diesem Grund wird in diesem Projekt der Analyseprozess des Trainingsdatensatzes fokussiert. Der Testdatensatz bleibt unbeachtet. Die gewählten Visualisierungstechniken dienen der Unterstützung der Untersuchung des Trainingsdatensatzes auf gewählte Eigenschaften. Ziel ist es anhand der Visualisierungen den Trainingsdatensatz auf Anomalien, Ausreißer, Verteilungen und verzerrten Werte zu untersuchen, sowie zu bereinigen, um folglich eine Unverzerrtheit dessen zu gewährleisten. Entsprechend ist das Baumdiagramm aufgrund der Möglichkeit hierarchische Daten darstellen zu können, ausgewählt worden. Durch dieses kann eine übersichtliche Darbietung der Verteilung der Automarken erreicht werden.

Die Verwendung des Scatterplots wird durch dessen Eigenschaft der Simplizität begünstigt. Im Hinblick auf die Zielgruppen des Projektes, kann fachliche Expertise nicht vorausgesetzt werden. Entsprechend bietet der Scatterplot eine leicht verständliche Visualisierung von Daten, welche auf Grundlage seiner Beschaffenheit dennoch einen erheblichen Mehrwert gegenüber Daten in tabellarischer Form bietet.

Die Zusammenhänge im System der Parallelen Koordinaten sind komplexer und bieten mehr Raum für Vergleichbarkeit von Eigenschaften. Mittels dieser Visualisierungen können multidimensionale Daten im zweidimensionalen Raum dargestellt werden. Mithilfe der interaktiven Funktion besteht der Vorteil des Vergleiches von mehr als 2 Attributen. In Kapitel wird ein detaillierterer Bezug zur Umsetzung und Auswertung der Visualisierungstechniken genommen.

# Daten

Die Grundlage dieses Projekts bilden Datensätze der Webseite „Kaggle“. [10] Die Spezialisierung dieser Plattform liegt auf den Sektoren Datenanalyse und Maschinellen Lernen. Eine Community stellt auf diesem Kontext basierende Datensätze öffentlich für die Weiterverarbeitung zur Verfügung. *Used Cars Price Prediction* des Nutzers Avi Kasliwalist einer dieser und findet im vorliegenden Projekt Anwendung. Als Lokation Avi Kasliwals ist die indische Stadt Noida im Bundesstaat Uttar Pradesh angegeben. Genauere Beschreibungen zum Datensatz sind von der Organisation Co-Learning Lounge angegeben. Diese erläutern die Relevanz des Gebrauchtwagenhandels und erklären das daraus resultierende Bedürfnis der Optimierung des Validierungsprozesses von Gebrauchtwagen. Additional verweisen sie auf optionale Aufgabenstellungen diesbezüglich. Co-Learning Lounge ist eine Plattform indischen Ursprungs, welche einer Community Raum für gemeinsames Lernen, Zusammenarbeiten, Innovieren und Mentorieren bietet. Die Ausrichtung ist in diesem Zusammenhang auf die Schwerpunkte künstliche Intelligenz, Blockchain, Big Data, Automatisierung und Technik ausgerichtet. [4]

Das letzte Update der Datensätze wurde vor einem Jahr vorgenommen. Entsprechend kann von einer Aktualität der Daten ausgegangen werden. Die Originaldatensätze sind ein Trainingsdatensatz namens *train.csv* und ein Testdatensatz namens *test.csv.* Zusätzlich existiert eine *.xlsx* Datei namens *dictionary.xlsx*, welche Auskunft über Inhalte der verwendeten Variablen innerhalb der Datensätze gibt. Die Originaldatei des Trainingsdatensatzes verfügt über 13 Spalten mit insgesamt 6019 Datensätzen. Das original des Testdatensatzes enthält 12 Spalten mit 1234 Datensätzen. Innerhalb der Datensätze wird sich auf die Informationen *Name* (deutsch: Name), *Location* (deutsch: Ort), *Year* (deutsch: Jahr), *Kilometers\_Driven* (deutsch: gefahrene Kilometer), *Fuel\_Type* (deutsch: Kraftstofftyp), *Transmission* (deutsch: Schaltgetriebe), *Owner\_Type* (deutsch: Vorbesitzer), *Mileage* (deutsch: Kilometerstand), *Engine* (deutsch: Hubraum), *Power* (deutsch: Pferdestärken), *Seats* (deutsch: Sitze), *New\_Price* (deutsch: neuer Preis) und *Price* (deutsch: Preis) bezogen. Der *dictionary.xlsx* Datei sind charakterisierende Informationen der Spaltennamen zu entnehmen. Der *Name* gibt die Marke und das Model des Fahrzeugs aus. Die *Location* deklariert den Ort an welchem das Auto verkauft wurde oder zum Kauf angeboten wird. Die Auflage beziehungsweise das Baujahr des Modells wird durch die Variable *Year* wiedergegeben. *Kilometers\_Driven* geben Auskunft über die gefahrenen Kilometer des Fahrzeugs durch Vorbesitzer. *Fuel\_Type* deklarierte die Art des Kraftstoffes mit welchem der Gebrauchtwagen angetrieben wird. Die Variable *Transmission* definiert die Art des Schaltgetriebes. Mittels des *Owner\_Types* wird beschrieben wie viele Vorbesitzer das Fahrzeug in Benutzung hatten. *Mileage* gibt den Verbrauch an. Wie viel Kilometer können mit einem Liter Kraftstoff gefahren werden. Das Hubraumvolumen des Motors definiert sich über die Variable *Engine. Power* deklariert die Leistungsfähigkeit des Motors. Die Anzahl der Sitze wird durch *Seats* ausgegeben. *New\_Price* gibt den Preis des Neuwagens des Models aus. Die *Price* Variable enthält Informationen des Gebrauchtwagens.

Aus bereits beschriebenen Gründen stellt das Ziel dieser Arbeit die Visualisierung und Auswertung des Trainingsdatensatzes dar. Dies bezüglich wird im Folgenden ausschließlich Bezug auf diesen genommen. Generalisierend kann zunächst davon ausgegangen werden, dass die im Datensatz enthaltenen Informationen bezüglich der Validierung des Preises eines Gebrauchtwagens von Relevanz sind. Ob die Datenmenge und Aussagekraft der Informationen ausreichend für Maschinelles Lernen sind, ist dem Kapitel 7 dieser Arbeit zu entnehmen.

Um eine optimierte Datengrundlage für die Umsetzung der Visualisierungen bereitzustellen, mussten die Daten einer Analyse unterzogen werden. Auffällig in diesem Zusammenhang war es, dass die Spalten *Name, Location, Year, Kilometers Driven, Fuel Type, Transmission, Owner Type* und *Price* vollständig gedeckt sind. Die Informationen in den Spalten *Mileage, Engine, Power* und *Seats* sind lückenhaft. Additional ist liegt eine Diversität der Einheiten der Spalte Mileage vor. Generalisierend kann die Auswahl der Einheiten aller Spalten als semioptimal betrachtet werden. Die Kilometer per Liter wurden auf die Einheit *kmpl* angeglichen, der Hubraum des Fahrzeugs wird allgemein gültig in Liter angegeben, für die Stärke des Motors wird die Einheit Pferdestärken verwendet (PS). Additional hat eine Umrechnung des Preises des Gebrauchtwagens in Euro stattgefunden. Eine detailliertere Betrachtung des Modifizierungsprozesses der Daten kann Kapitel 2.2. entnommen werden.

## Technische Breitstellung der Daten

Die Datensätze werden als CSV-Datei bereitgestellt. [10] Eine CSV-Datei ist definiert als Comma-Seperated-Value. Entsprechend werden einzelne Datenwerte durch Komma getrennt aufgelistet. Vorteil dieses Dateityps ist die Kompatibilität der Integration mit einer Vielzahl von Programmen. Im Kontext des vorliegenden Projektes wurde der Trainingsdatensatz zur Bearbeitung in Microsoft Excel eingefügt und als modifizierter Datensatz exportiert.

Die technische Bereitstellung der Daten erfolgt über ein GitHub Repository. [11] In diesem wurde das Visualisierungsprojekt gehostet. In diesem sind die Daten der Visulisierungen Scatterplot und Parallele Koordinaten als *.csv* Datei im *data* Ortner vorzufinden. Die Baumhierarchie arbeitet mit dem Dateiformat *.json* ***(J****ava****S****cript* ***O****bject* ***N****otation).* Die Quelldaten liegen im Ortner *Quelldaten.* Alle durch Selektion oder Modifizierung veränderten Daten sind dem Ortner *AufbereiteteDaten* zu entnehmen.

## Datenvorverarbeitung

Bereits in Kapitel 2 wurde Bezug zur Notwendigkeit der Modifizierung des Trainingsdatensatzes genommen. Diese Anpassung basiert in drei Schritten. In Abfolge werden die Daten gesichtet, bearbeitet und überführt. Eine detailliertere Prozessbeschreibt folgt in diesem Kapitel.

Das Ziel des Sichtungsprozesses der Daten war es die Werte in ein gut zu bearbeitendes Format zu übertragen. Entsprechend fand die Konvertierung der Trainingsdatensatz *train.csv* in eine Exceldatei Anwendung. Die Umsetzung dessen zielt auf eine übersichtliche tabellarische Darstellung der Datenwertpaare, sowie den Zugriff auf den Umfang der Excelfunktionen ab.

Im zweiten Schritt wurden die Daten aufbereitet, sodass deren reibungslose Verarbeitung durch den Einsatz in Code gewährleistet werden konnte. Entsprechend wurden in erster Instanz, aufgrund fehlender Aussagekraft sowie Vergleichbarkeit, Datenwertpaare entfernt, welche Nullwerte oder Lücken in den Spalten *Mileage, Engine, Power* und *Seats* enthalten. Zusätzlich wurde die Spalte *New\_Price* auf Basis lückenhafter Informationen, sowie fehlender Irrelevanz zum vorliegenden Kontext, entfernt. In zweiter Instanz wurden alle englischen Bezeichnungen im Spaltenkopf ins deutsche Äquivalent überführt. Folgend fand eine Generalisierung der Einheiten innerhalb einer Spalte und die damit verbundene Umrechnungen Anwendung. Da der vorliegende Bericht in der deutschen Sprache verfasst ist, wurden zusätzlich alle englischen Informationen zur besseren Verständlichkeit ins deutsche übersetzt.

Um die Werte innerhalb der Zellen als Zahl identifizieren zu können, bestand zunächst die Notwendigkeit alle Einheitsbezeichnungen innerhalb der Zellen zu entfernen. Nachfolgend wurden alle Werte der Einheit *km/kg* in der Spalte *Mileage* in *kmpl* umgerechnet. Der Umrechnungsfaktor beträgt 1.4. Alle Ergebnisse wurden auf zwei Dezimalstellen gerundet. Zusätzlich fanden Modifikationen in der Spalte *Engine* Anwendung. Alle Werte waren in *CC* angegeben. Es ist davon auszugehen, dass *CC* die Einheit Kubikzentimeter wiedergibt. Da diese gebräuchliche Anwendung im Kontext zu Angaben des Hubraums ist. Die Einheitsangaben wurden entfernt um den Inhalt der Zelle als zahl definieren zu können. Alle Informationen über die *Power* des Fahrzeugs sind im Originaldatensatz mit der Einheit *bhp* beschrieben. *Bhp* steht für Brake horsepower. 1 bhp entspricht 1.0139 Pferdestärken (PS). Da PS im europäischen Raum als handelsübliche Einheit für die Leistung des Motors eines Fahrzeugs betrachtet wird, fanden Umrechnungen in PS statt. Zusätzlich musste die Ziffer innerhalb der Zelle als Zahl definiert werden. Die Spalte *Price* war in der Währung *Lakh* angegeben. *Lakh* gilt als südasiatischen Zahlenwort für einhundertausend. [6] Insbesondere im Zusammenhang mit der Angabe der indischen Währung *Indische Rupien (INR)* findet dieser Wortlaut Anwendung. Entsprechend bezeichnet 1 *Lakh, einhunderttausend* Indischen Rupien*.* Zusätzlich liegt der Wechselkurs von *Indischen Rupien* in *Euro* bei 83,5096 INR zu 1€. [7] Entsprechend dieser Annahmen fand die Umrechnung der Werte innerhalb der Spalte *Price* statt. Wiederholt hat eine Entnahme der Einheitsangaben innerhalb der Zellen stattgefunden, um den Wert in dieser als Zahl definieren zu können. Die Ergebnisse wurden auf ganze Zahlen ohne Dezimalstellen gerundet. Erklärend für diesen Sachverhalt ist der Vergleich der Preisangaben zu erfolgreichen Gebrauchtwagenplattformen, wie *mobile.de* oder *autoscout24.de*. [3, 14]Diesen Webseiten zu entnehmen sind Preisangaben in ganzen Zahlen, ohne Dezimalstellen. Dementsprechend wurde diese Richtlinien übernommen und bezüglich der Preisangaben in den Visualisierungen angewandt.

Für die Darstellung der Baumhierarchie war zusätzlich eine Umwandlung des Dateiformats von *.csv* in *.json* notwendig. In diesem Zusammenhang besteht die Möglichkeit die Daten aus Microsoft Excel in eine *.json* zu exportieren. Mittels dieser Anwendung liegt allerdings keine Baumstruktur der Daten vor. Diese gilt als unabdingbar für Anordnung eines Baumes. Folglich bestand die Notwendigkeit der manuellen Bearbeitung der exportierten Exceldatei mit Visual Studio Code. Demgemäß konnte eine Baumhierarchie manuell erzeugt werden. Die Daten wurden dafür nach Gebrauchtwagen, Marken und final nach Modellen geordnet. Im direkten Vergleich der Vorverarbeitung der Daten der 3 Visualisierungstechniken, kann die Vorverarbeitung der Baumhierarchie als aufwendigste betrachtet werden. Die entsprechende .*json* Datei ist im Ordner *AufbereiteteDaten* unter dem Namen *LocationCar.json* vorzufinden.

Zuletzt wurde eine Spalte ID für jeden Gebrauchtwagen hinzugefügt. Diese verhilft der eindeutigen Identifizierung des Datensatzes. Jeder ID-Wert ist eindeutig innerhalb der Tabelle. Auf diese Weise besteht bei alleiniger Angabe der ID, die Möglichkeit für den Nutzer bei Bedarf ergänzende Informationen über das Objekt anhand der Datentabelle zu erfahren.

Im dritten Schritt der „Überführung der Daten“, bestand das Ziel die Werte für den Programmcode lesbar zu machen. Entsprechend wurden alle innerhalb von Microsoft Excel modifizierten Daten, in ein *.csv* Format zurückgeführt. In diesem Prozess lag der Fokus darauf, dass die Daten durch Kommas getrennt, sowie als Dezimaltrennzeichen ein Punkt Verwendung fand. Im folgenden Kapitel wird detaillierter auf die Entwicklung der Visualisierungen eingegangen.

# Visualisierungen

Wie bereits verdeutlicht, erfolgt die Darstellung des Trainingsdatensatzes von Gebrauchtwagen anhand von 3 Visualisierungstechniken. In erster Instanz wird mit Hilfe eines Baumdiagrammes die quantitative Verteilung der vorhandenen Datensätze analysiert. Anschließend wird ein Scatterplot erzeugt, um den Vergleich zweier Attribute des Gebrauchtwagens zu ermöglichen. Ziel ist es, dass der Nutzer diese Attribute interaktiv wählen kann. Mitunter besteht allerdings die Notwendigkeit mehr als zwei Attribute miteinander zu vergleichend. Entsprechend wurde die dritte Visualisierung der Parallelen Koordinaten gewählt, welch einen solchen Vergleich gewährleistet. Folglich können die Zusammenhänge von mehr als zwei Eigenschaften betrachtet werden. Die Reihenfolge dessen soll wiederholt für den Nutzer wählbar sein.

Im folgenden Kapitel wird detaillierter auf die Anforderungen an die Visualisierungen, sowie deren technische Umsetzung eingegangen. Des Weiteren werden im Verlauf

## Analyse der Anwendungsaufgaben

Die Visualisierungen sollen eine Unterstützung für in Kapitel 1.2 erläuterte Zielgruppen darstellen. Vordergründing sollen sie verhelfen die Informationen der Gebrauchtwagen darzustellen, sodass bei dem Nutzer ein Zugewinn an Erkenntnissen besteht, welcher ohne die Visualisierung nicht stattgefunden hätte.

Im Kontext eines Baumdiagrammes lassen sich Datensätze in Hierarchie Ordnen und anzeigen. Dies ermöglicht eine übersichtliche Darstellung der Daten nach bestimmten Gesichtspunkten. Im vorliegenden Beispiel soll die Eigenschaft der hierarchischen Darstellung genutzt werden, um eine Verteilung der Datensätze analysieren zu können. Entsprechend ist das Wurzelelement als Gebrauchtwagen definiert und gliedert sich in die Marke und das Model des Automobils. Anhand dessen lassen sich quantitative Rückschlüsse über die Repräsentativität konkreter Fahrzeugtypen treffen.

Hingegen liegen die Vorteile des Scatterplots sowie der parallelen Koordinaten in der Möglichkeit des Erfassens von Trends und Anomalien innerhalb der Daten. Das Ziel des Scatterplots ist es, Zielgruppen eine Gegenüberstellung zweier Attribute zu ermöglichen. Dieser Ansatz erbringt den Vorteil, dass Zusammenhänge dieser Eigenschaften analysiert werden können. In diesem Kontext können beispielsweise Fragen beantwortet werden, wie: *Wie verhält sich der gefahrene Kilometerstand eines Gebrauchtwagens zu dessen Preis? Sind Gebrauchtwagen mit mehr PS grundsätzlich preisintensiver?* Insbesondere bei der Analyse statistischer Zusammenhänge gilt die Darstellung zweier Faktoren in zweidimensionalen Koordinatensystemen als effektives Hilfsmittel und findet in der Praxis Anwendung. Aufgrund der Simplizität des Systems kann ein leichtes Verständnis des Scatterplots beim Nutzer vorausgesetzt werden. Das Wählen der Attribute, welche miteinander verglichen werden sollen, kann über zwei Buttons selbstdefiniert gesteuert werden. Entsprechend werden die gewählten Eigenschaften auf der X-Achse, sowie Y-Achse dargestellt. Kausal folgt auf einen Wechsel der Attribute, eine Veränderung der Achsen und des Plots. Die Darstellung eines Scatterplots basiert vorerst ausschließlich auf einer Punktwolke. In diesem Fall kann nicht nachvollzogen werden, welcher Punkt welches Fahrzeug darstellt. Aus diesem Grund wurde eine „Hover“-Funktion der Namensanzeige und Farbänderung programmiert. Insbesondere bei der Analyse großer Datenmengen erweist sich diese Funktion als vorteilhaft. Basierend der Überschneidungen einer Vielzahl diskreter Variablen werden, tendiert die Darstellung unübersichtlich zu wirken. Mit Hilfe einer solchen „Hover“-Funktion kann diese Problematik umgangen werden.

Basierend der Grundlagen des Scatterplots kann das System der Parallelen Koordinaten als Erweiterung dessen betrachtet werden. Anstelle zweier Attribute tritt der Vergleich mehrerer Eigenschaften, welche gleichermaßen interaktiv ausgewählt werden können. Additional zu den bereits erwähnten Vorteilen können zusätzlich Nutzdefinierte Anforderungen gefunden und befriedigt werden. Beispielsweise können Antworten auf Fragen wie folgend gefunden werden. *Welcher Gebrauchtwagen ist Baujahr 2010, hat einen geringen Kilometerstand, 100 PS und kostet weniger als 10.000€?* Schlussfolgernd ist abzuleiten, dass diese Darstellung über die zwei Dimensionen des Scatterplots hinausgeht. Die gewählten Attribute werden miteinander durch Linien verbunden. Entsprechend sind der Darstellung Zusammenhänge zu erkennen. Wiederholt kristallisiert sich das Problem der Unübersichtlichkeit durch Überlagerung diskreter Werte heraus. Die Lösung dessen besteht, ähnlich wie beim Scatterplot, in der Implementierung der „Hover“-Funktion, welche mittels Namensänderung und Farbanzeige zu mehr Verständnis beiträgt. Anhand dessen kann der Nutzer die Ausgabe von Informationen eines konkreten Fahrzeugs durch die Kennzeichnung einer Linie entlang der Parallelen Koordinaten angezeigt bekommen. Grundlegend ist das System der Parallelen Koordinaten deutlich komplexer als das des Scatterplots. Zusätzlich verfügt diese Visualisierungstechnik nicht über einen gleichwertigen Bekanntheitsgrad. Entsprechend kann das Wissen der Zielgruppen über dessen Interpretation nicht vorausgesetzt werden. Aus diesem Grund bietet sich eine Beilage von Erklärung über die Interpretation der Parallelen Koordinaten an.

## Anforderungen an die Visualisierungen

In Kapitel 3.1 wurde bereits Bezug genommen, welche Ziele die gewählten Visualisierungen adressieren. Aus diesen Zielstellungen kann eine Reihe von Anforderungen abgeleitet werden.

Innerhalb des Baumdiagrammes werden starre Hierarchien repräsentiert. Anhand dessen kann der Anwender strukturierte Informationen über die Marken, sowie Modelle verfügbarer Gebrauchtwagen entnehmen. Entsprechend können Ableitungen zu deren Verteilungsverhalten getroffen werden.

Ziel der zweiten Visualisierung ist es zwei Attribute der Gebrauchtwagen miteinander ins Verhältnis zu setzen. Stilistisches Mittel in diesem Zusammenhang ist der Scatterplot. Um die Werte in einem Scatterplot zuordnen, sowie ablesen zu können ist eine fachgerechte Beschriftung der X-Achse und Y-Achse notwendig. Dies impliziert einerseits die ordnungsgemäße Angabe der Namen an den Achsen. Andererseits existieren Zahlenwerte zu den Attributen. Diese müssen an den Achsen mit entsprechenden Einheiten aufgeführt werden und folglich die Achse im Verhältnis dieser Zahlenwerte einteilen. Zusätzlich wird im Kontext des Scatterplots auf individuelle Interessenvertretung abgezielt. Schlussfolgend soll es den Nutzer möglich sein, selbstständig eine Auswahl der Attribute entsprechend seinem Interessengebiet auszuwählen. Die Umsetzung erfolgt über Buttons, welche beim Anklicken die gewünschte Eigenschaft der geforderten Achse zuordnen. Zusätzlich wird an die Visualisierungstechnik die Anforderung der Übersichtlichkeit an die Visualisierungstechnik gestellt. Die Ausgabe in Form einer Punktewolke kann diese Anforderung nicht erfüllen. Entsprechend wurde eine „Hover“-Funktion programmiert, welche dem Nutzer ermöglicht beim Anklicken eines Punktes dessen informatorischen Bezug zu dem konkreten Fahrzeug auszugeben. Neben diesen Informationen färbt sich der Punkt rot, für eine bessere Gesamtübersicht im Scatterplot. Schlussfolgernd konnte die Anforderung der Übersichtlichkeit mit Hilfe der integrierten Funktion gelöst werden.

Bei der Visualisierung der Parallelen Daten liegt der Fokus darauf, Zusammenhänge zwischen mehr als zwei Daten zu verdeutlichen. Kausal der Mehrdimensionalität besteht die Annahme der Möglichkeit erhöhter Unübersichtlichkeit. Entsprechend liegt die Beschränkung auf dem Vergleich von maximal 4 Attributen. Zusätzlich wurde zur Steigerung des Übersichtlichkeitsaspektes die „Hover“-Funktion eingeführt.

Zusammenfassend besteht die Notwendigkeit der Anforderungen in der Umsetzung der Übersichtlichkeit erhobener Parameter. Diese dient der Optimierung der Ableitung infrastruktureller Maßnahmen. Kausal kann dadurch eine Komplexitätsreduktion der Datenmenge erzielt werden, welche simplifizierende Auswirkung im Aufzeigen von Zusammenhängen hat. Additional besteht die Anforderung eines Interaktionsraumes für die Zielgruppen. Dieser bietet durch individuelle Wahl vergleichender Attribute, die Möglichkeit der persönlichen informatorischen Bedürfnisbefriedigung. Kausal muss eine Diversität innerhalb der Attribute vorliegen, um die Abdeckung unterschiedlicher Informationssektoren, sowie die Heterogenität der Darstellungen zu gewährleisten. Weiterhin muss der Wahrheitsgehalt der Datenwerte und deren Übereinstimmung innerhalb der Visualisierungen sichergestellt werden.

## Präsentation der Visualisierung

Dieser Abschnitt dient der Repräsentation der zugrundeliegenden Visualisierungstechniken der Baumhierarchie, des Scatterplot und der Parallelen Koordinaten.

### Visualisierung Eins

Als erste Visualisierung wurde die Baumhierarchie gewählt. Diese dient der Ordnung der Gebrauchtwagendaten und wird aus diesem Grund in erster Instanz vorgestellt. Sie gibt den Anwender einen Überblick über verfügbare Automarken und deren Modelle und lässt in diesem Zusammenhanf Rückschlüsse über deren Verteilung zu. Der Graph eines Baumes enthält keine geschlossenen Pfade. Folglich besteht die Möglichkeit der Darstellung einer Monohierarchie. Grundelemente des Baumes sind Knoten und Kanten. Knoten definieren die durch die Hierarchie vorgegebenen Objekte. Jeder Knoten stellt in diesem Kontext eine Entscheidungsmöglichkeit dar. Eine Kante deklariert die Beziehung zweier Knoten und stellt deren Verbindung dar. Abbildung 2 zeigt einen Ausschnitt eines umfangreichen Baumdiagrammes.

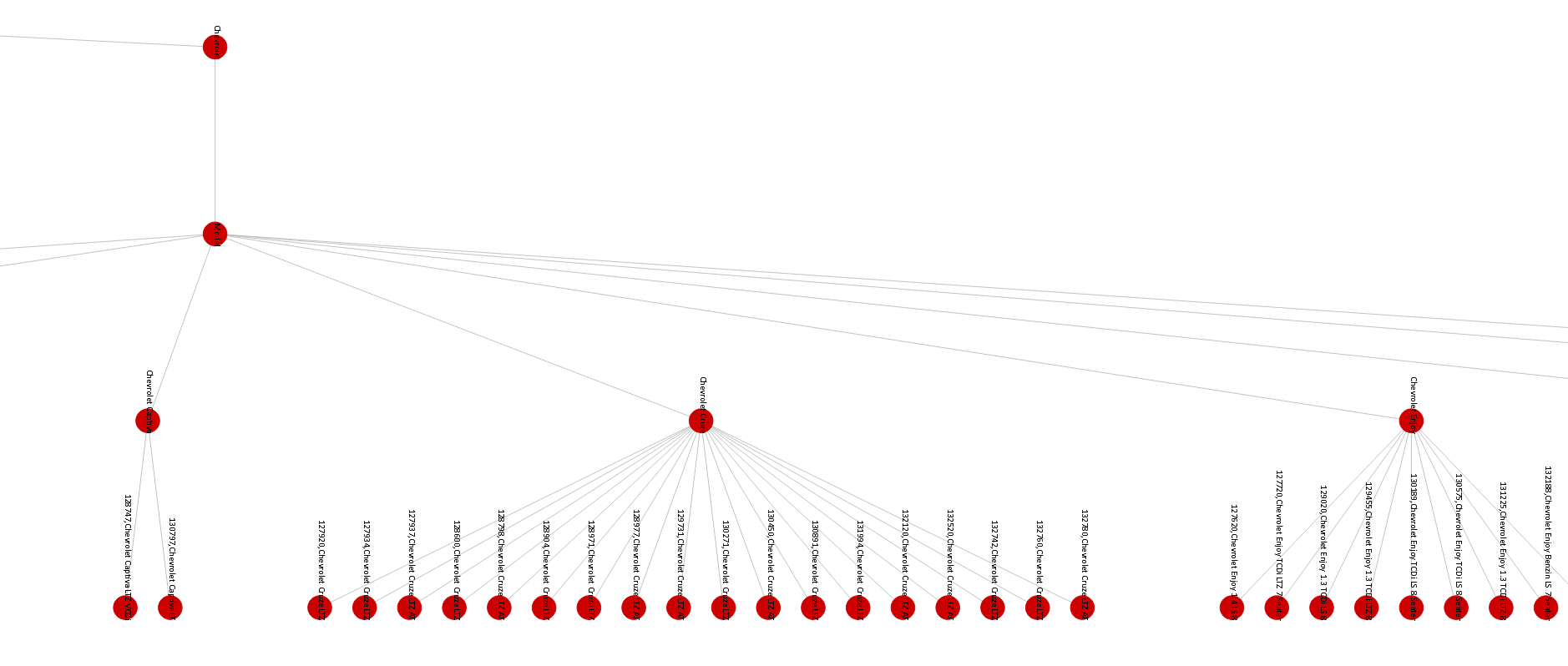


Abbildung 2: Auszug aus der Baumhierarchie (Quelle: eigene Darstellung)

Das Baumdiagramm besitzt 3 Ebenen. Die Wurzel entspricht der Quelle der Marken der Gebrauchtwagen. Auf der Zwischenebene liegt eine namentliche Unterscheidung dieser vor. Die unterste Ebene differenziert die Modelle innerhalb der Marken. Aus Perspektive der Übersichtlichkeit wurden Knoten rot eingefärbt. Entsprechend konnten die Anforderungen, der hierarchischen Strukturierung und Übersichtlichkeit der Darstellung, umgesetzt werden. Die Visualisierung basiert auf dem Datenformat einer *.json* Datei. Diese wurde manuell angelegt und ist unter *LocationCar.json* im Repository vorzufinden. Die Eigenschaft dessen ist ein starres Dateiformat. Entsprechend wurde auf die Interaktionsmöglichkeit des Anwenders verzichtet. Trotz einer Vorfilterung der Daten auf die Eigenschaft der Fahrzeugmarke, sowie dessen Models erscheint der Datensatz als sehr umfangreich. Um die Wahrung der Übersichtlichkeit gewährleisten zu können, wurde auf weitere Unterteilungen, wie Pferdestärken oder Baujahr verzichtet. Dennoch können diese Informationen den Visualisierungstechniken des Scatterplots, sowie der Parallelen Koordinaten entnommen werden. Alternativ zur Darstellung des Baumdiagrammes werden diverse Graphenanwendungen gesehen. Diese ermöglichen eine Mehrfachzuweisung zu Knoten. Entsprechend kann eine Zunahme der Komplexität der Darstellung verzeichnet werden. Die Wahrscheinlichkeit einer fehlerhaften Wahrnehmung der Hierarchien steigt. Insbesondere bei einer erhöhten Anzahl von Objekten. Kausal kann eine steigende Komplexität mit einer sinkenden Motivation der Wissbegier des Modells beim Anwender einhergehen. Schlussfolgernd wurde sich für die Verwendung der Baumhierarchie entschieden.

### Visualisierung Zwei

Als zweite Visualisierungstechnik wurde der Scatterplot gewählt. Das Ziel dessen ist die Darstellung zweier Eigenschaften der Gebrauchtwagen. Die Realisierung dessen ist Abbildung 3 zu entnehmen.

Abbildung 3: Scatterplot (Quelle: eigene Darstellung)

Zusehen ist ein zweidimensionales Koordinatensystem, welches durch die Achsen *x* und *y* bestimmt ist. Jede dieser Achsen entspricht den Zahlenwerten der verwendeten Eigenschaft. Die Punkte innerhalb des Plots geben die Koordinatenwerte beider verwendeter Attribute wieder. Basierend des individuellen Informationsbedürfnisses hat der Nutzer die Möglichkeit zur Auswahl der wiederzugebenden Attribute an den jeweiligen Achsen. Entsprechend werden 7 Eigenschaften pro Achse bereitgestellt, um den Informationsbedarf decken zu können. Der Informationsumfang inkludiert Daten über das Baujahr, den Kilometerstand, die Pferdestärken, den Preis in Euro, die Anzahl der Sitze, die Reichweite pro Liter, sowie des Hubraumes des Fahrzeuges. Über die obigen implementierten Buttons kann eine dynamische Auswahl dessen erfolgen. Auf Grundlage des Umfangs des Datensatzes steigt das Ausmaß der Punktewolke. Die Wirkung der entstehenden Darstellung ist folglich unübersichtlich. Um den Zielgruppen weitere Informationen bereitstellen zu können und somit eine Steigerung der Übersichtlichkeit der Visualisierung zu ermöglichen, wurde eine „Hover“-Funktion implementiert. Diese färbt einen Punkt, bei Bewegung mit der Maus über diesen. Erweiternd werden Informationen über diesen Punkt ausgegeben. Im Detail entsprechen diese Informationen dem Fahrzeugtyp, sowie Angaben zu den gewählten Eigenschaften. Schlussfolgernd konnten die eingangs aufgestellter Anforderungen bezüglich des Scatterplots umgesetzt werden. Additional fand anhand der Möglichkeit der individuellen Achsenbestimmung, eine Erweiterung dessen statt. Diese befähigt den Anwender den Schwerpunkt über Darstellungsinhalte eigenständig wählen zu können.

Als Alternative zu einem Scatterplot können die Darstellungsformen der Zeitreihen oder Liniendiagramme in Betracht gezogen werden. Zeitreihen stellen Zeitverläufe von Daten dar. Im Datensatz der Gebrauchtwagen ist das Attribut *Baujahr* als zeitliche Ausprägung zu verstehen. In Anbetracht des Hintergrundes der Zielgruppen, insbesondere der Zielgruppe des Data Scientist, fehlt es der ausschließlichen Betrachtung der zeitlichen Komponenten des Trainingsdatensatzes an notwendiger Relevanz. Die Darstellung von Zusammenhängen stellt hingegen eine Bedeutung dar., worauf im folgenden Kapitel näher eingegangen wird. Ein Liniendiagram besitzt die Fähigkeit Attribute gegenüberzustellen. Im Gegensatz zum Scatterplot würden die Fahrzeuge nicht als Punktewolke dargestellt werden. Stattdessen findet die Verbindung über eine Linie Anwendung. Anhand Abbildung 3 ist den Punkten kein stetiger Verlauf entlang der x-Achse nachzuweisen. Vielmehr finden Überlagerungen statt. Schlussfolgernd kann eine Darstellung anhand eines Liniendiagrammes als ungeeignet, aufgrund von Unübersichtlichkeit, gewertet werden. Im Gegensatz dessen werden die Vorteile der Eindeutigkeit, sowie der Simplizität des Scatterplots deutlich. Diese unterstreichen dessen Expressivität, sowie Effektivität und begründen die Wahl dieser Visualisierungstechnik.

### Visualisierung Drei

Aufgrund seiner Beschaffenheit liegt die Limitation des Scatterplots in der Darstellung zweier Attribute. Eine Gegenüberstellung von mehr als zwei Eigenschaften ist auf Basis dieser Visualisierungstechnik nicht möglich. Ein effektives Mittel diese Art von Zusammenhängen graphisch darzustellen, stellt die Technik der Parallelen Koordinaten dar, wie in Abbildung 4 zu sehen.

Abbildung 4: Auszug aus der Baumhierarchie

Auf dessen Basis können mehrdimensionale Daten simultan über vertikale Achsen dargestellt werden. Dabei entfällt die X-Achse. Die Zahlenwerte einer Ausprägung werden an entsprechender Achse abgetragen und mit Hilfe von Polygonlinien miteinander verbunden. In diesem Zusammenhang repräsentiert jede Linie ein Fahrzeug. Der Vorteil dieser Technik ist die Möglichkeit der simultanen Darstellung von 4 relevanten Eigenschaften, welche eine Signifikanz in der Analyse des Trainingsdatensatzes aufweisen. Infolgedessen kann nicht, wie im Scatterplot, ausschließlich der Zusammenhang zweier Eigenschaften, sondern die Kausalität von vier Attributen in Bezug genommen werden. Äquivalent zur Umsetzung des Scatterplots, verfügt die Darstellung der Parallelen Koordinaten über die Funktion der individuellen Auswahl der gewünschten Attribute. Der Anwender hat die Wahl zwischen den Eigenschaften des Baujahrs, des Kilometerstandes, der Pferdestärken, des Preises in Euro, der Anzahl der Sitze, der Reichweite pro Liter, sowie des Hubraumes des Fahrzeuges. Entsprechend kann der Anwender die Darstellung nach seinem Bedürfnis verändern, die Achsen beliebig tauschen und folglich individuelle Zusammenhänge erzeugen. Weiterhin wurde in der Darstellung der Parallelen Koordinaten die „Hover“-Funktion implementiert. Bei der Bewegung des Mauszeigers über eine der Polygonlinie, verfärbt sich diese in der Farbe Rot. Zusätzlich erscheinen am Kopf der Darstellung Informationen über die Identität, des durch die Linie repräsentierten, Fahrzeugs, sowie dessen Werte der Ausprägungen der gewählten Attribute. Weiterhin steigert eine gute Achsenbeschriftung die Übersichtlichkeit des Models. Basierend des Datenumfangs entstehen Linienüberlappungen. Die Intensivierung der Linienfarbe Grau lässt auf diese Kontiguität schließen. Folglich können Rückschlüsse auf Indentifikationsmuster gezogen werden. Schlussfolgernd wurden die Anforderungen an die Visualisierungstechnik erfüllt.

Neben den Parallelen Koordinaten können durch die Techniken der Datentinte, Projektionen, Selektionen, Sternkoordinaten mehrdimensionale Daten graphisch umgesetzt werden. Das Model der Datentinte reduziert die Inhalte auf eine geringe Anzahl von Kernelemente, welche zuvor mit Hilfe von Berechnungen identifiziert wurden. Folglich entfällt die notwendige Weitsicht der Darstellung, welche zum Ausschluss von dessen Anwendung führt. Projektionen, sowie Selektionen bilden den mehrdimensionalen Raum ab. Insbesondere bei großen Datenmengen ist ein Verlust der Übersichtlichkeit dieser Modelle zu verzeichnen und erschwert die Lesbarkeit für den Anwender. Sternkoordinaten fokussieren die Abbildung mehrdimensionaler Daten in 2D oder 3D, mit sternförmiger Achsenanordnung. Im direkten Vergleich wurde allerdings die Visualisierungstechnik der Parallelen Koordinaten in den Punkten Expressivität, Nachvollziehbarkeit und Übersichtlichkeit trotz großer Datenmengen als geeigneteres stilistisches Mittel eingestuft und fand folglich im vorliegenden Projekt Anwendung.

## Interaktion

Mit Hilfe von Interaktionen kann eine optimale Interessenvertretung seitens des Anwenders gewährleistet werden. Entsprechend wurde in den Visualisierungstechniken Scatterplot und Parallele Koordinaten die Umsetzung von Interaktionsmöglichkeiten für den Anwender angestrebt.

Im Scatterplot verfügt der Nutzer die Option über eine Interaktion mit der Datenquelle. Mit Hilfe von Buttons kann dieser aus einer Auswahl von Eigenschaften, die für ihn signifikanteste wählen und einer Achse zuordnen. Dadurch kann der Fokus individuell auf gewünschte Sachverhalte gelegt werden. Kausal entspricht der Informationsgehalt der Darstellung den Bedürfnisstrukturen des Anwenders. Insbesondere Data Scientists können von diesem Faktor profitieren. Ihnen unterliegt die Möglichkeit Zusammenhänge von Eigenschaften zu zentrieren, welche Auffälligkeiten aufweisen und denen es folglich mehr Beachtung bedarf. Um eine Steigerung der Übersichtlichkeit zu erzielen, sowie simultan die Anwendungsfreundlichkeit zu erhöhen wurde die „Hover“-Funktion integriert. Basierend einer aktiven Bewegung mit Maus des Nutzers über einen Punkt im aufgespannten Koordinatensystem, verfärbt sich der Punkt in der Farbe Rot und es werden Informationen über die Identität des Fahrzeuges, sowie die Inhalte dessen Attribute ausgegeben. Ergänzend werde die exakten Datenwerte angezeigt, welche anhand der Achseneinteilungen zu Ungenauigkeiten führen könnten. Äquivalent verhält sich die Interaktion im Kontext der Parallelen Koordinaten. Erweiternd bietet diese Technik den Raum für eine individuelle Zuordnung von Attribut und Achse. Entsprechend soll dies die Motivation des Nutzers dahingehend beeinflussen, interessante Kombinationsmöglichkeiten und folglich deren Sichtweise umzusetzen. Beispielsweise ist eine Kombination denkbar, welche eine Überlappung von Polygonlinien provoziert, um Trends identifizieren zu können. Da die Baumhierarchie mit dem Dateiformat einer starren JSON-Datei arbeitet, muss diese Technik auf eine Interaktion verzichten. Allerdings ist zu bemerken, dass auf Grundlagen der hierarchischen Strukturen das Visualisierungsziel bereits erfüllt ist.

Additional wurden die drei Visualisierungen als HTML-Seite verlinkt. Mit Hilfe einer Vernetzung der HTML-Seiten untereinander, entsteht Interaktionsspielraum für den Anwender. Mit Hilfe von konfigurierten Buttons kann der Nutzer zwischen den drei Visualisierungen und der Startseite wechseln und somit entsprechend seinen Bedürfnissen, die für ihn optimale Darstellung der gewünschten Zusammenhänge wählen.

Zusammenfassend ist festzustellen, dass die Ausprägung der Interaktion nach Visualisierungstechnik variieren. Die größte Ausprägung unterliegt dem Scatterplot und den Parallelen Koordinaten. Ursächlich dafür kann die Notwendig der Interaktion zur Erreichung des Visualisierungsziels gesehen werden. Dieses betrifft die Bereitstellung eines hohen Informationsgehaltes, die Gewährleistung der Analyse aus dem Zusammenspiel von Attributen, sowie die Möglichkeit des Attribute zueinander ins Verhältnis zu setzen und Rückschlusse daraus ziehen zu können.

# Implementierung

­­Das folgende Kapitel soll Aufklärung über die Umsetzung der Visualisierungen im Elm-Code geben. Die Verwaltung des Projektes erfolgt im folgenden Git-Repository: [milu1992/Elm-Projekt-Used-Cars---main (github.com)](https://github.com/milu1992/Elm-Projekt-Used-Cars---main). In diesem liegen vier Ortner: *src, final, data* und *bericht.* Im Source-Codebefinden sich die Ortner *Test* und *Zwischenstände*. Diese sind während des Programmierprozesses entstanden und spiegeln die Entwicklung des Projektes wider. Ersterer enthalt alle Skripte, welche zum Testen der Visualisierungen, sowie der Datensätze genutzt wurden. Diese fanden beispielsweise Verwendung bei Testversuchen des Einlesens von Daten, sowie beim Erstanlegen eines Scatterplots. Im Ortner *Zwischenstände* befinden sich fertig funktionierende Codes, welche noch keine Ordnung im Code aufweisen. Im Ortner *final* wurde der Code der fertigen Darstellungen abgelegt. Der *data*-Ortner enthält Quelldaten und Aufbereitete Daten. Unter Quelldaten sind alle Originaldatensätze von der Website *Kaggle* zu verstehen. Aufbereitete Daten enthalten alle modifizierten Datensätze als CSV- oder JASON-Dateien. Um den Anwender einen leichten Zugriff auf die drei umgesetzten Visualisierungstechniken zu gewährleisten, wurde eine *HTML-*Seite erstellt. Diese ermöglicht den Aufruf der drei Darstellungen. In der Umsetzung wurde sich einer *index.html* Startseite. Diese kann aus dem persönlichen Github-Repository eine Website darstellen. Anschließend wurde über den Terminalbefehl *elm make* zu jeder Visualisierung eine *HTML-*Seite erstellt. Basierend einer entsprechenden Vernetzung untereinander, wird eine Interaktion ermöglich. Alle verwendeten *HTML-*Seiten sind im gleichnamigen Ortner vorzufinden.

Es folgt eine detaillierte Betrachtung des Quellcodes. Um eine Übersichtlichkeit dessen zu gewährleisten, wurde eine einheitliche Struktur des Codes anvisiert. Zu Beginn jedes Codes werden alle notwendigen *Module importiert*. Die Schnittmenge aller drei Visualisierungen bildet in diesem Zusammenhang der Zugriff auf die Packages „*Html*“ sowie „T*ypedSVG*“. Anschließend folgt die *main-*Funktion. Diese ermöglicht eine Datenspeicherung basierend eines Records. Folgend ist die Definition der „*type“* und *„type alias*“. Im Anschluss erfolgt der Prozess der *Datenbeschaffung* und des Dateneinlesens. Die *Initialisierung* bildet die Basis für das Durchlaufen des *Decoders.* Im weiteren Verlauf werden Funktionen definiert. Diese dienen der Datenverarbeitung, sowie der Darstellung optischer Präferenzen. Im finalen Block werden die Funktionen *subscriptions,* gefolgt von dem *view* und *update*. Die *view* Funktion veranlasst die Darstellung der Visualisierung, das *update* die Veränderung der Achsen entsprechend den hinterlegten Attributausprägungen. Zur Steigerung der Übersichtlichkeit wurde der Quellcode mit erklärenden Kommentaren versehen.

Die informatorische Grundlage für dieses Projekt bilden die Übungen, sowie Vorlesungen des Moduls: Information Retrieval und Visualisierung mit Schwerpunkt Information Visualisierung, gelehrt von PD Dr. Alexander Hinneburg an der Martin-Luther Universität Halle-Wittenberg. [13]

Die Umsetzung der Baumhierarchie basiert auf den Grundlagen der Übung 10. Diese bat einen nahezu vollständigen Modell Code für die erste Visualisierungstechnik. Entsprechende Abänderungen mussten ausschließlich in der Datenbereitstellung vorgenommen werden. Folglich bedurfte es des Einfügens des Pfades zur JASON-Datei. Entsprechend entsprach das Programmieren des Codes dieser Visualisierungstechnik wenig Herausforderung. Im Gegenzug dessen gestaltete sich die Bereitstellung der Daten als deutlich anspruchsvoller und zeitintensiver. Ursächlich dafür war die Notwendigkeit der Umwandlung der CSV-Dateien in ein JASON-Format. Für diesen Prozess konnte keine zufriedenstellende, automatisierte Lösung gefunden werden. Schlussfolgernd wurde die JASON-Datei manuell in Visual Studio Code angefertigt.

Bezüglich der zweiten Visualisierungstechnik sind insbesondere sind die erste sowie dritte Übung von erhöhter Bedeutung. Diese bedienten die Themen des Scatterplots und dessen Optionen der Interaktionen. Der Code für die Darstellung des Scatterplots konnte zu großen Teilen aus Übung eins entnommen werden. Der Unterschied zu den Übungen lag im Projekt in der Verarbeitung einer CSV-Datei, welche zuvor importiert werden musste. Ein Beispiel für diesen Prozess konnte Übung 8 entnommen werden. Dieser entsprang die Vorlage des CSV-Decoders, welche in der Darstellung des Scatterplots, sowie der Paralallen Koordinaten vorausgesetzt wurde. Eine identische Übernahme war nicht möglich. Vielmehr mussten die Funktionen entsprechend der Darstellung um die dazustellenden Werte erweitert werden. Um eine Interaktion für den Anwender zu ermöglichen, wurde sich für die Verwendung von Buttons entschieden. Durch deren Betätigung soll eine Veränderung des Attributes an x- oder y-Achse des Scatterplots hervorgerufen werden. Dessen Realisierung stellte eine Herausforderung dar. Probleme entstanden bei Anpassungen der Programmgliederung. Dementsprechend wurden Informationen aus der dritten und achten Übung entnommen und in der Update-Funktion integriert. Ergänzend verhalf die Recherche der Elm-Dokumentation zur Umsetzung. [5] Neben Anpassungen der Update-Funktion setzt die Umsetzung der gewünschten Aktivität der Buttons weitere Datenverarbeitungen, sowie die Verknüpfung der Daten mit den Buttons voraus. Entsprechend fanden unter anderem Anpassungen am CSV-Decoder statt.

Für die Visualisierungstechnik der Parallelelen Daten wurden Informationen der Übung sieben benötigt. Der Quellcode dieser Übung speigelte die Vorlage für die Programmierung der Parallelen Koordinaten wieder. Zusätzlich konnten gewonnen Erkenntnisse aus dem Erstellen des Codes des Scatterplots übernommen werden. Beispielsweise fand der CSV-Decoder des Scatterplots Anwendung im Code der Parallelen Koordinaten.

Zusammenfassend stellte rückblickend die Erweiterung des CSV-Decoder die größte Herausforderung dar. Dieser musste die Anforderung umsetzen mehr als vier Spalten simultan verarbeiten zu können. Contra produktiv in diesem Zusammenhang ist die Eigenschaft, dass Tupel in Elm maximal drei Items annehmen. Schlussfolgernd musste der Decoder neu geschrieben werden.

# Anwendungsfälle

In diesem Kapitel werden Anwendungsfälle der drei Visualisierungen vorgestellt. Basierend des gegebenen inhaltlichen Hintergrunds der gewünschten Nutzung der Daten, wird bei der Analyse der Fokus auf die Zielgruppe des Data Scientist gelegt. Anwendungsfälle für die zwei weiteren erwähnten Zielgruppen werden ausschließlich komprimiert betrachtet. Wie bereits im Kapitel 1.1 näher erläutert, ist der Trainingsdatensatzes in Bezug zum maschinellen Lernen von enormer Bedeutung. Der Trainingsdatensatz ist ein Datensatz mit Beispielen, die für das Lernen der Muster und Zusammenhänge in den Daten verwendet wird. Die Anpassung der Gewichte des Algorithmus wird über den Trainingsdatensatz antrainiert. Dies bedeutet der Algorithmus lernt aus den Daten. Entsprechend muss der Trainingsdatensatz Unverzerrtheit, sowie einen ausreichenden Umfang aufweisen. Folglich muss dieser vor dessen Verwendung analysiert und evaluiert werden. Die drei umgesetzten Visualisierungstechniken sollen diesen Prozess erleichtern. Entsprechend wird im Folgenden jede Darstellung den Trainingsdatensatz aus unterschiedlicher Perspektive analysieren.

## Anwendung Visualisierung Eins

Die Vorteile des Baumdiagrammes sind in zwei wesentlichen Aspekten zu sehen. Zunächst bietet es dem Data Scientist die Möglichkeit einen anschaulichen Überblick über die verfügbaren Marken der Gebrauchtwagen, sowie deren Modelle zu bekommen. Diese Eigenschaft die im allgemeinen Kontext als hilfreich, kann aber bezüglich der Analyse des Trainingsdatensatzes nicht als essenziell eingestuft werden. Im Gegensatz dazu ist es von Vorteil, basierend der strukturierenden Eigenschaft der Baumhierarchie, eine Verteilung der Daten erkennen zu können. Im Detail ist die Information bekannt, dass auf Basis des Trainingsdatensatzes die Anpassung der Gewichtung des Algorithmus erfolgt und sich daraus letztlich Muster der Daten ergeben. Eine höhere Anzahl an Inputdaten führt zu spezifischeren Ergebnissen im Output. Dementsprechend ist eine gleichmäßige Verteilung in den Daten bezüglich der Marke und des Modells der Fahrzeuge von Vorteil. In Anbetracht des Baumdiagrammes ist diese gleichmäßige Verteilung nicht gegeben.

Abbildung 5: Scatterplot welcher die Eigenschaften Preis und Körper gegenüberstellt (Quelle: eigene Darstellung)

## Anwendung Visualisierung Zwei

Abbildung 6: Gegenüberstellung der Eigenschaften Körper, Gerbstoffe, Süße und Säuregehalt mithilfe der Parallelen Koordinaten (Quelle: eigene Darstellung)

## Anwendung Visualisierung Drei

Abbildung 7: Ausschnitt mit Asiatischen Weinen aus der Baumhierarchie (Quelle: eigene Darstellung)

# Verwandte Arbeiten

# Zusammenfassung und Ausblick

# Anhang

## Git Historie

Literatur

1. (2020) Kurz gefasst: Künstliche neuronale Netze in 3 Minuten. https://blog.fintechcube.com/kuenstliche-neuronale-netze-in-3-minuten/. Zugegriffen: 29. November 2021

2. (2021) Parallele Koordinaten. https://de.wikipedia.org/wiki/Parallele\_Koordinaten. Zugegriffen: 29. November 2021

3. Auto Scout24. https://www.autoscout24.de/auto/gebrauchtwagen/. Zugegriffen: 05. Dezember 2021

4. Co-Learning Lounge. https://colearninglounge.com/. Zugegriffen: 01. Dezember 2021

5. Czaplicki E Elm Documentation. https://elm-lang.org/docs

6. dictionary.com. https://www.dictionary.com/browse/lakh. Zugegriffen: 03. Dezember 2021

7. finanzen.net. https://www.finanzen.net/devisen/euro-indische\_rupie-kurs. Zugegriffen: 24. November 2021

8. Goodfellow I, Bengio Y, Courville A (Hrsg) (2018) Deep Learning. Das umfassende Handbuch. Grundlagen, aktuelle Verfahren und Algorithmen, neue Forschungsansätze. MITP-Verlag

9. Gumm,H.,Peter Bäume. Bäume, Binärbäume, Traversierungen, abstrakte Klassen, Binäre Suchbäume, Balancierte Bäume, AVL-Bäume, Heaps, Heapsort, Priority queues. https://www.mathematik.uni-marburg.de/~gumm/Lehre/SS07/PraktischeInformatikII/Folien/08%20Baeume.pdf. Zugegriffen: 20. November 2021

10. Kasliwal A Used Cars price Prediction. Predict the price of an unknown car. Build your own Algo for cars 24!! https://www.kaggle.com/colearninglounge/used-cars-price-prediction. Zugegriffen: 29. November 2021

11. Lindner M-L (2021) Elm-Projekt-Used-Cars---main. https://github.com/milu1992/Elm-Projekt-Used-Cars---main. Zugegriffen: 19. Dezember 2021

12. Lovelace A (1842) Notes upon L. F. Menabrea’s “Sketch of the Analytical Engineinvented by Charles Babbage”

13. martin-Luther Universität Halle-Wittenberg Alexander Hinnebrug. https://users.informatik.uni-halle.de/~hinnebur/

14. mobile.de. https://www.mobile.de/. Zugegriffen: 05. Dezember 2021

15. Nantasenamat C (2020) The Data Science Process. A Visual Guide to Standard Procedures in Data Science. https://towardsdatascience.com/the-data-science-process-a19eb7ebc41b. Zugegriffen: 30. November 2021

16. Traeger M, Eberhart A, Geldner G, Morin AM, Putzke C, Wulf H, Eberahrdt,L.,H.,J. (2003) Künstliche neuronale Netze. Theorie und Anwendungen in der Anästhesie, Intensiv- und Notfallmedizin. Der Angesthesist:1055–1061

17. Yi M A Complete Guide to Scatter Plots. Chartio. https://chartio.com/learn/charts/what-is-a-scatter-plot/. Zugegriffen: 27. November 2021