HAW HAMBURG

MASTER THESIS
Mareile Beernink genannt Konjer

Tabellarische Datensynthese mit Fokus auf Differential Privacy unter Verwendung von Generative Adversarial Networks

FAKULTÄT TECHNIK UND INFORMATIK
Department Informatik

Faculty of Engineering and Computer Science
Department Computer Science

HOCHSCHULE FÜR ANGEWANDTE WISSENSCHAFTEN HAMBURG
Hamburg University of Applied Sciences

Mareile Beernink genannt Konjer

Tabellarische Datensynthese mit Fokus auf Differential Privacy unter Verwendung von Generative Adversarial Networks

Masterarbeit eingereicht im Rahmen der Masterprüfung
im Studiengang *Master of Science Informatik*
am Department Informatik
der Fakultät Technik und Informatik
der Hochschule für Angewandte Wissenschaften Hamburg

Betreuender Prüfer: Prof. Dr. Ulrike Steffens
Zweitgutachter: Prof. Dr. Björn Gehlsen

Eingereicht am: 08. Dezember 2023

Mareile Beernink genannt Konjer

Thema der Arbeit

Tabellarische Datensynthese mit Fokus auf Differential Privacy unter Verwendung von Generative Adversarial Networks

Stichworte

Tabellarische Datensynthese, Differential Privacy, Generative Adversarial Networks, Künstliche Intelligenz, Maschinelles Lernen, DaFne, Smart City

Kurzzusammenfassung

Eine der häufigsten Herausforderungen bei der Einführung und Nutzung von KI-Anwendungen liegt in der Beschaffung der Trainingsdaten. Einerseits mangelt es an ausreichend vielen und qualitativ hochwertigen Daten, andererseits können sensible Daten aufgrund der Gefahr des Privatsphärenverlusts nicht genutzt werden. Mit dem Ziel schützenswerte Daten nutzbar zu machen, ohne die Privatsphäre zu gefährden, setzt sich die Thesis mit der Implementierung von Differential Privacy (DP) in Generative Adversarial Networks (GAN) auseinander. Im Rahmen des Forschungsprojektes DaFne wird konkret nach einem geeigneten DP-GAN gesucht, das zum einen die Eigenschaften der realen Daten abbildet und zum anderen die Privatsphäre schützt. Untersucht werden die Ursprungsmodelle DPGAN & PATE-GAN und die fortgeschrittenen Modelle CTAB-GAN+ & DP-CGANS anhand von zwei Datensätzen unterschiedlicher Komplexität sowie unter Berücksichtigung verschiedener Größen des Privatsphären Budgets. Zusammenfassend überzeugt das CTAB-GAN+ bezüglich Trainingsdauer, Datenqualität sowie Privatsphärenschutz. Insbesondere übertrifft es die anderen Modelle durch eine hohe Datenqualität auch bei geringem Privatsphären Budget sowie durch seine Leistung bei der Generierung hoch-dimensionaler Daten.

Mareile Beernink genannt Konjer

Title of Thesis

Synthesis of differential private tabular data using Generative Adversarial Networks

Keywords

Tabular Data Synthesis, Differential Privacy, Generative Adversarial Networks, Artificial Intelligence, Machine Learning, DaFne, Smart City

Abstract

One of the most common challenges in the introduction and utilization of AI-applications lies in the procurement of training data. On the one hand there is a lack of sufficient and high-quality data, while on the other hand sensitive data cannot be used due to the risk of privacy loss. With the aim of rendering data both valuable and safeguarded while maintaining privacy, the thesis explores the integration of Differential Privacy (DP) into Generative Adversarial Networks (GAN). The DaFne research project is specifically looking for a suitable DP-GAN that on the one hand maps the properties of real data and on the other hand protects privacy. The original models DPGAN & PATE-GAN as well as the advanced models CTAB-GAN+ & DP-CGANS are analyzed using two data sets of different complexity and considering different sizes of the privacy budget. In summary, CTAB-GAN+ is convincing in terms of training duration, data quality, and privacy protection. In particular, it outperforms the other models due to its high data quality even with a low privacy budget, and its performance in generating high-dimensional data.

Inhaltsverzeichnis

Abbildungsverzeichnis

Tabellenverzeichnis

1 Einleitung

Seit einigen Jahren wird der Begriff Künstliche Intelligenz (KI) nicht mehr einzig von der Wissenschaft geprägt. Weltweit erfolgen Investitionen in diverse KI-Anwendungen in unterschiedlichen Wirtschaftszweigen [57]. Auch nimmt die Präsenz von KI im privaten Bereich durch Anwendungen wie Chatbots (z.B. ChatGPT) und die Integration von KI in Alltagsgegenständen wie Smartphones, Smart-Home-Geräten oder Zahnbürsten stetig zu.

Zu den häufigsten Schwierigkeiten bei der Ausweitung von KI-Initiativen zählt, vor der eigentlichen Implementierung der KI-Technologien, die Beschaffung der Trainingsdaten für Machine Learning (ML)-Modelle. Das Management von KI-bezogenen Risiken, Vorschriften wie die DSGVO und die Skepsis der Bevölkerung gegenüber KI verstärken die Herausforderungen [57]. Das mit dieser Thesis im Zusammenhang stehende Forschungsprojekt Data Fusion Generator für die Künstliche Intelligenz (kurz: DaFne) stellt sich der Problematik des begrenzten Zugangs zu ausreichenden und qualitativ hochwertigen Daten. Hierzu können tabellarische Daten auf Basis verschiedener Datensätze fusioniert und mittels Regeln oder Reproduktion generiert werden.

Ein nicht zu vernachlässigender Grund für den Mangel an nutzbaren Daten besteht in der Gefahr eines Verlusts an Privatsphäre. In den vergangenen Jahren ereigneten sich zahlreiche große Datenschutzverletzungen [88]. Der Wettbewerb „Netflix-Prize“ zählt zu einem der bekanntesten Vorfälle, bei dem Filmbewertungen ohne persönliche Identifikatoren veröffentlicht wurden und dennoch Forscher unter Inanspruchnahme zusätzlicher Datenquellen die Verfasser der Bewertungen zu 99% identifizieren konnten [61]. Auch weitere Beispiele beweisen, dass traditionelle Anonymisierungs-Techniken gegenüber Datenverknüpfungen invalide sind und darüber hinaus die Genauigkeit von Modellen senken. Ohne die Gewissheit, dass die zu anonymisierenden Daten ausschließlich im Kontext der eigenen Datenverarbeitung genutzt werden, kann mit herkömmlichen statistischen Verfahren keine garantierte Privatsphäre sichergestellt werden.

Um dennoch ohne Bedenken mit Daten arbeiten zu können, die personenbezogene Informationen beinhalten, besteht die Alternative, Datenverteilungen mittels Generativer ML-Modelle zu lernen und auf Basis dessen neue Daten zu synthetisieren. In Kombination mit der Differential Privacy (DP) [23] können Datenschutzbudgets vorab festgelegt und während der Evaluation überprüft werden. Als eine Art von Generativen ML-Modellen werden in dieser Thesis Generative Adversarial Networks (GAN) [34] untersucht.

Mit dem Ziel schützenswerte Daten ohne Verlust von Privatsphäre nutzbar zu machen, beschäftigt sich die Thesis daher mit der Integration von Differential Privacy in Generative Adversarial Networks. Im Kontext des Forschungsprojekts DaFne wird konkret nach einem DP-GAN gesucht, das sowohl die Eigenschaften der realen Daten abbildet als auch die Privatsphäre schützt. Daher lautet die Forschungsfrage der Thesis:

Welches Generative Adversarial Network eignet sich für eine adäquate Synthese sensibler tabellarischer Daten unter Berücksichtigung von Differential Privacy?

Zur Beantwortung der Forschungsfrage erfolgt eine Evaluation von drei Teilaspekten:

1. Wie viel Zeit benötigt das Modell für die Synthese von Daten? (**Performance**)
2. Inwiefern entsprechen die Eigenschaften der vom Modell generierten Daten denen der Trainingsdaten? (**Datenqualität**)
3. Wie sicher sind die vom Modell generierten Daten gegenüber Angriffen? (**Privatsphärenschutz**)

Um sich der Forschungsfrage zu nähern, werden zunächst in Kapitel 2 die Grundlagen zu GANs, DP sowie Praktiken zur Integration von DP in GANs erklärt. Darauf aufbauend werden in Kapitel 3 die zu evaluierenden Modelle vorgestellt sowie eine Auswahl an erweiterter Literatur aufgezeigt. Bevor in Kapitel 5 die Evaluationsergebnisse präsentiert werden, wird der Aufbau der Experimente beschrieben (Kapitel 4). Mit Hilfe eines simulierten Fallbeispiels sowie einem komplexeren realen Datensatz werden die ausgewählten Modelle bezüglich der drei genannten Teilaspekte unter Berücksichtigung verschiedener Privatsphären Budgets bewertet. Auf die Evaluation folgt die Diskussion in Kapitel 6. Zusätzlich zur kritischen Bewertung einzelner Ergebnisse wird die Forschungsfrage beantwortet und die Modellauswahl für DaFne getroffen. Abschließend erfolgt in Kapitel 7 eine Zusammenfassung der Thesis inklusive Ausblick.

2 Grundlagen

Im folgenden Kapitel werden die für das Verständnis der Thesis erforderlichen Grundlagen vermittelt. Beginnend mit einer konzentrierten Einführung in das übergreifende Themengebiet Machine Learning werden anschließend die beiden Hauptkomponenten Generative Adversarial Networks (GAN) und Differential Privacy (DP) vorgestellt. Darauf aufbauend folgt die Darstellung der zwei am häufigsten verwendeten Methoden zur Integration von DP in GANs: „Differentially Private Stochastic Gradient Descent“ sowie „Private Aggregation of Teacher Ensembles“.

2.1 Machine Learning (ML)

Im Bereich der Künstlichen Intelligenz (KI) hat sich über die letzten zwei Jahrzehnte vor allem das Teilgebiet Machine Learning (ML) etabliert. Das übergeordnete Ziel der KI, menschliche Verhaltensmuster zu imitieren, wird im ML durch das Training von Modellen basierend auf Daten erzielt [66]. Im Gegensatz zu den anfänglichen Vorgehensweisen von KI müssen keine konkreten Verhaltensregeln definiert und keine Wissensdatenbank aufgebaut werden. Die heute am stärksten ausgeprägte ML-Variante nennt sich Deep Learning (DL). Die Inspiration für diese Art des Lernens stammt aus der Funktionsweise und dem Aufbau biologischer Neuronaler Netze [8].

2.1.1 Aufbau Neuronaler Netze

Häufig wird ein Künstliches Neuronales Netz (engl. artificial neural network, ANN) für Aufgabenbereiche wie Klassifikation, Regression, Bildverarbeitung oder Generierung von Daten eingesetzt. Explizit zielt das Neuronale Netz darauf ab, komplexe Muster und Zusammenhänge in vorhandenen Daten zu lernen [8]. ANNs bestehen aus Input, Hidden sowie Output Schichten, wobei jede Schicht Knoten bzw. Neuronen enthält, die durch Kanten mit Neuronen anderer Schichten verbunden sind [83].

Abbildung 2.1 zeigt ein Fully Connected Netz. Jedes Neuron des Netzes ist mit allen Neuronen der vorherigen sowie nachfolgenden Schichten verbunden. Die Tiefe des Netzes bezieht sich auf die Anzahl der Schichten des gesamten Netzes und beträgt in Abbildung 2.1 vier (1x Input, 2x Hidden, 1x Output). Ein ANN mit mehr als einer Hidden Schicht nennt sich Deep Neuronal Network (DNN). Moderne Neuronale Netze besitzen zumeist mehrere Hidden Schichten, tausende bis Millionen Neuronen sowie hunderte Millionen von Verbindungen [8].

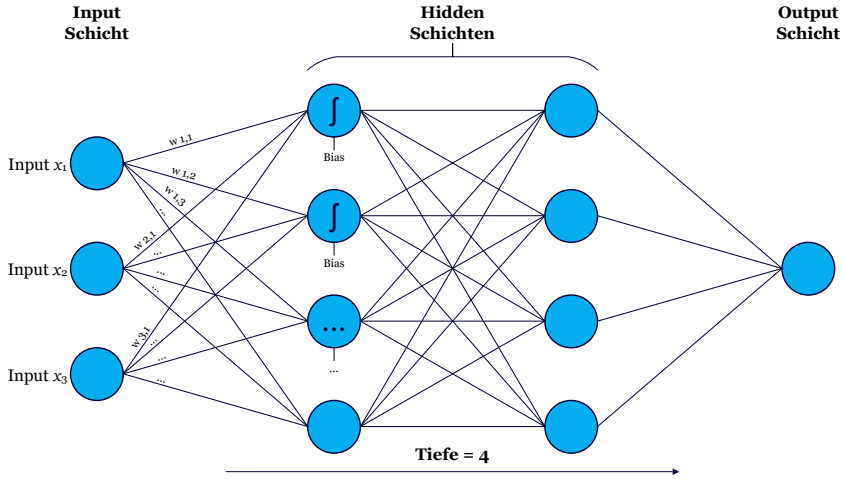

Abbildung 2.1: Aufbau eines Neuronalen Netzes der Tiefe 4 (eigene Darstellung)

2.1.2 Funktionsweise Neuronaler Netze

Optimierer wie Stochastic Gradient Descent (SGD) trainieren Neuronale Netze, indem sie das Ziel verfolgen eine Kostenfunktion zu minimieren. Inkrementell werden die Gradienten aller lernbaren Parameter über die Backpropagation berechnet und anschließend genutzt, um die Parameter des Netzes zu aktualisieren [83]. Typische lernbare Parameter sind Gewichte an den Verbindungen zwischen Neuronen, Bias-Werte von Neuronen sowie Dropout-Wahrscheinlichkeiten. Dropout erlaubt eine vorübergehende zufällige Deaktivierung einzelner Verbindungen zwischen Neuronen, um ein Overfitting zu reduzieren.

Auch die unterschiedlichen Aktivierungsfunktionen der Neuronen besitzen einen großen Einfluss auf die Funktionalität des Netzes. In der Regel handelt es sich bei ihnen um nichtlineare Funktionen, die das Lernen von komplexen Zusammenhängen ermöglichen. Häufig verwendete Aktivierungsfunktionen sind die Sigmoid, ReLU (Rectified Linear Unit)

und Tangens Hyperbolicus (tanh) Funktionen [33]. Durch ihre unterschiedlichen Verläufe und Wertebereiche eignen sie sich für verschiedene Problemstellungen und besitzen unterschiedliche Herausforderungen.

Im Wesentlichen sind folgende Schritte relevant für den **Lernprozess** des Netzes [33]:

1. **Forward pass**: Nachdem die Eingabedaten über die Input Schicht im Netzwerk integriert sind, beginnt sukzessiv (Schicht für Schicht) die Datenpropagation.
   a) Jedes Neuron multipliziert die Gewichte w seiner Eingangsverbindungen mit seinen Eingangsdaten x, summiert alle Eingänge n auf und addiert sie mit einem Bias-Wert b. Der Bias-Wert ist ein zusätzlicher Parameter in jedem Neuron. Er ermöglicht durch eine Verschiebung der Aktivierungsfunktion Vorhersagen besser an die Eingangsdaten anzupassen.

$$z = \sum_{i=1}^{n} w_i \cdot x_i + b \tag{2.1}$$

   b) Im Anschluss wird die Aktivierungsfunktion des Neurons auf das Ergebnis z angewendet.

   Das Gesamtergebnis bildet dann wiederum ein Eingangsdatum für verbundene Neuronen der folgenden Schicht. Sobald die Output Schicht des Neurons erreicht ist, erfolgt die Berechnung des Fehlers.
2. **Loss calculation**: Mithilfe einer definierten Kostenfunktion wird der Fehler des Netzes bestimmt. Berechnet wird er unter Berücksichtigung des Vergleichs von Zielwert zur tatsächlichen Ausgabe des Netzes. Der Fehler gibt Aufschluss über die Leistung bzw. die Genauigkeit der Vorhersagen des Modells.
3. **Backward pass**: Mit dem Ziel Parameter zu finden, die den Fehler minimieren, wird bei der Rückpropagation die Kostenfunktion partiell nach allen lernbaren Parametern des Neuronalen Netzes abgeleitet. Unter Inanspruchnahme der Kettenregel werden die Gradienten der Gewichte und Bias-Werte basierend auf den Ableitungen von Kostenfunktion, Aktivierungen vorheriger Schichten und gewichteter Summe der Eingänge (inkl. Bias-Wert) berechnet.
4. **Parameter update**: Gemäß der Lernrate α nehmen die im Backward Pass errechneten Gradienten $\nabla_\theta \mathcal{L}(\theta)$ Einfluss auf die Aktualisierung der einzelnen Parameter θ. Da beim Lernen ein Minimierungsproblem gelöst werden soll, wird die Berechnung mit dem negativen Gradienten durchgeführt.

$$\theta^{new} = \theta^{old} - \alpha \cdot \nabla_\theta \mathcal{L}(\theta^{old}) \tag{2.2}$$

Mit dem Update aller Parameter endet eine Lern-Iteration. Iterationen durch den gesamten Trainingsdatensatz werden zu einer Epoche zusammengefasst [83]. Je nach Komplexität eines Modells, Größe des Datensatzes und Anforderungen an das Problem ist eine geeignete Epochenanzahl unterschiedlich groß. Die Anzahl an Iterationen pro Epoche ist ebenfalls von mehreren Faktoren abhängig. Zusätzlich zur Größe des Datensatzes können auch unterschiedliche Optimierungsverfahren die Anzahl beeinflussen. Im Gegensatz zum SGD, bei dem die Parameter pro Datum aktualisiert werden, aktualisiert das Mini-Batch Gradienten Verfahren beispielsweise die Parameter pro Datensatzgruppe (Batch).

Ergänzend zum SGD gibt es weitere fortgeschrittene Optimierer. Monumentum zum Beispiel sorgt durch die Berücksichtigung vorheriger Gradienten für eine beschleunigte Suche des Minimums der Kostenfunktion. Andere gängige Optimierer wie Ada Grad, RMS Prop und Adam verwenden adaptive Lernraten, um das Modelltraining zu verbessern. Jeder Parameter kann seine eigene Lernrate besitzen, die in Abhängigkeit von vorherigen Ergebnissen optimiert werden kann [33].

2.1.3 Generative Modelle

Im Fokus dieser Arbeit stehen Generative Adversarial Networks (GAN). GANs gehören zu den Generativen Modellen. Im Gegensatz zum Deskriptiven Modell, das Wahrscheinlichkeiten abschätzt, lernt das Generative Modell p_{model} eine Wahrscheinlichkeitsverteilung p_{data} [27]. Mit dem Ziel, die Trainingsdaten (*Beobachtungen*) möglichst gut abzubilden, wird p_{data} während des Modelltrainings optimiert. Eine weit verbreitete Methode zur Anpassung der Parameter von generativen Modellen ist die Maximum likelihood estimation (MLE) [32]. Bei der MLE werden die Modellparameter θ derart geschätzt, dass die Wahrscheinlichkeit (Likelihood) der Beobachtungen maximiert wird.

Die Lernvarianten zur Umsetzung der Methode unterscheiden sich in der Darstellung und Approximation des Likelihoods. Im Wesentlichen wird zwischen expliziten und impliziten Dichtemodellen unterschieden. Die expliziten Dichtemodelle definieren eine konkrete Likelihood-Funktion $p_{model}(x;\theta)$, die maximiert werden kann. Im Anschluss an das Training erfolgt in einem zweiten Schritt die Generierung der Daten. Implizite Dichtemodelle stellen keine Wahrscheinlichkeitsverteilung bereit. Sie ermöglichen hingegen eine indirekte Interaktion mit p_{data} durch die Erstellung unmittelbarer Stichproben. Das Training der Generative Adversarial Networks beruht auf dem impliziten Modell. Neue Daten werden direkt aus der durch das Modell repräsentierten Verteilung generiert [32].

2.2 Generative Adversarial Network (GAN)

Ian Goodfellow et. al [34] publizieren 2014 die erste Forschungsarbeit zu Generative Adversarial Networks. Zusätzlich zur Vorstellung des Aufbaus und der Funktionsweise von GANs, generieren die Autoren erste Bilddateien und diskutieren Vor- und Nachteile des vorgeschlagenen Frameworks. Aufbauend auf dieser Grundlage wurden GANs in den letzten Jahren auf unterschiedliche Weise optimiert und für verschiedene Datenarten modifiziert. Neben unterschiedlichen Lernverfahren haben sich weitere Komponenten als Ergänzung zur Grundstruktur als vorteilhaft erwiesen [45]. Während GANs zu Beginn insbesondere für die Generierung von Bilddaten [35, 64, 80, 43] verwendet wurden, werden sie heute u.a. auch für die Generierung von Texten [52, 94], Zeitreihendaten [39, 24] und Musik [60, 91, 36] genutzt. Die Synthese tabellarischer Daten liegt im Fokus dieser Thesis.

Das Fehlen von geeigneten Daten und die hohe Performance von GANs spiegeln sich zudem in der Menge an Publikationen sowie in ihrer weit verbreiteten Anwendung wider [35]. Insbesondere im Bereich der Medizin gibt es zahlreiche Implementierungen von GANs. Beispielsweise werden die Modelle für das Design von DNA-Strukturen [48], die Verarbeitung medizinischer Bilder [5, 50, 68, 84] oder die Nutzung privater Patientenakten [16] verwendet. In der Informatik unterstützen sie u.a. in den Gebieten Cybersecurity [42], Datenschutz [1, 7] oder auch Data Science [98, 86, 29]. Diese Arbeit legt den Schwerpunkt auf das Konzept Smart City. Die expliziten Anwendungsfälle werden in 4.4 beschrieben.

2.2.1 Aufbau und Funktionsweise von GANs

Auf Grundlage der Ursprungspublikation [34] werden folgend der Aufbau sowie die Funktionsweise von GANs vorgestellt. Der Name des Frameworks, Generative Adversarial Networks (dt. erzeugende gegnerische Netze), beschreibt seine wesentlichen Eigenschaften. Mit dem Ziel Daten zu erzeugen, trainieren zwei Neuronale Netze gegeneinander. Konkret trainieren ein Generator-Modell und ein Diskriminator-Modell gegeneinander.

Generator G entspricht dem impliziten Generativen Modell (siehe 2.1.3). Auf Basis eines Rauschvektors z und der gelernten Wahrscheinlichkeitsverteilung p_{data} werden die Daten G(z) erzeugt.

Diskriminator D handelt gleich einem binären Klassifikator. Als Input erhält dieser die vom Generator erzeugten Daten (fake data: G(z)) sowie einen originalen Trainingsdatensatz (real data: x). Ohne die Quelle der Eingaben zu kennen, lernt D die Daten nach Herkunft zu klassifizieren. In Abhängigkeit zu seiner Leistung erfolgt im Anschluss die Optimierung der eigenen Parameter sowie der Parameter des Generators.

Ziel des GANs ist erreicht, sobald der Generator den originalen Trainingsdatensatz so imitieren kann, dass der Diskriminator seine Eingangsdaten nicht mehr unterscheiden kann. Zu diesem Zeitpunkt wird das Training des Modells beendet und der Generator steht für die Generierung neuer Daten bereit. Der Diskriminator wird nicht weiter benötigt.

Je nach Dateneigenschaften und Modellarchitekturen kann es sinnvoll sein, die beiden Modelle einzeln oder unterschiedlich häufig zu trainieren. Demzufolge müssen der Generator und Diskriminator nicht immer hintereinander ausgeführt werden. Abbildung 2.2 visualisiert den Aufbau samt Ein- und Ausgaben eines GANs.

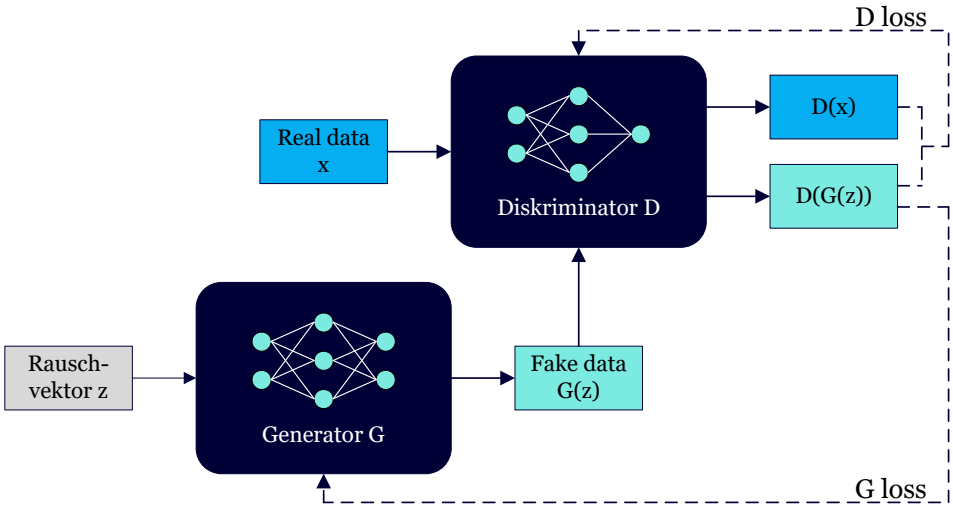

Abbildung 2.2: Aufbau eines Generative Adversarial Networks (in Anlehnung an [67])

Während Neuronale Netze üblicherweise nach einem Minimum suchen, hat der Diskriminator ein Maximierungsproblem zu lösen. Dementsprechend werden seine Parameter im Lernprozess nicht entlang des Gradientenabstiegs (Gradient Descent) neu berechnet, sondern mit Hilfe des Gradientenanstiegs (Gradient Ascent) optimiert. Beim Generator wird das übliche Gradientenabstiegs-Verfahren verwendet, um ein Minimierungsproblem zu lösen. Die Zielfunktion eines GANs verdeutlicht das Min-Max-Spiel zwischen Generator und Diskriminator:

$$V(D,G) = \mathbb{E}_{\mathrm{x} \sim \mathrm{p_{data}(x)}}[logD(x)] + \mathbb{E}_{\mathrm{z} \sim \mathrm{p_z(z)}}[log(1 - D(G(z)))] \tag{2.3}$$

Der erste Summand der Funktion misst die Wahrscheinlichkeit, dass der Diskriminator den realen Trainingsdatensatz korrekt identifiziert und folgend D(x) = 1 ergibt. Der zweite Summand misst die Wahrscheinlichkeit, dass der Diskriminator den generierten Datensatz korrekt identifiziert und folgend D(G(z)) = 0 bzw. 1 - D(G(z)) = 1 ergibt. Der Diskriminator zielt auf die Maximierung beider Teile ab, wohingegen der Generator ausschließlich Einfluss auf den zweiten Teil der Gleichung besitzt. Diesen versucht er zu minimieren, sodass der Diskriminator nicht mehr zwischen den realen und generierten Daten unterscheiden kann und D(x) = 0,5 entspricht.

2.2.2 Fortgeschrittene Architekturvarianten

Die Grundarchitektur von GANs hat sich über die vergangenen Jahre in verschiedene Richtungen weiterentwickelt. Entsprechend der in [67] vorgeschlagenen Taxonomie zur Gestaltung und Optimierung von GANs lassen sich die erweiterten Modellarchitekturen in sechs Kategorien unterteilen:

Bedingte Generierung Bei einem bedingten GAN erhält das Modell zusätzliche Informationen über die Trainingsdaten. Diese beeinflussen die Wahrscheinlichkeitsverteilung und unterstützen die Generierung der einzelnen Datengruppierungen. Ein weit verbreitetes bedingtes GAN ist das Conditional GAN (CGAN) [59]. Die Bedingung (engl. condition) c steht bei diesem Modell sowohl dem Generator als auch dem Diskriminator zur Verfügung.

Generator-Discriminator Paare Herkömmliche GANs zeigen aufgrund des Min-Max-Spiels eines einzelnen Generator-Diskriminator-Paars Probleme bei der Konvergenz, insbesondere bei komplexen Daten. Das Einführen von mehreren Generatoren

und Diskriminatoren kann dieser Herausforderung entgegenwirken und die Generierungsfähigkeit von GANs erhöhen. Folgend bestehen zusätzlich zum einfachen GAN auch Modelle, die mehrere Generatoren oder Diskriminatoren besitzen.

(a) **Training eines Generators**: Modelle dieser Art bestehen aus einem Generator-Diskriminator-Paar. Unterschiedliche Modifikationen wie z.B. die Zunahme von Modellschichten während des Trainings (vgl. ProcessGAN [47]) oder die Bereitstellung gebündelter Daten an den Diskriminator (vgl. PacGan [53]) optimieren die Geschwindigkeit und Stabilität des Trainings eines einfachen GANs.

(b) **Training mehrerer Generatoren**: Diese Modelle besitzen mehrere Generatoren. Die Anzahl der Diskriminatoren kann sich je nach Modelltyp unterscheiden. Während das MAD-GAN [30] beispielsweise aus einem Diskriminator und mehreren Generatoren besteht, verwendet das cGANs Framework [81] GAN-Ensembles. Mehrere GANs werden hintereinandergeschaltet und mit unterschiedlichen Teilen der Trainingsdaten trainiert.

(c) **Training mehrerer Diskriminatoren**: Diese Modelle setzen sich aus einem Generator sowie mehreren Diskriminatoren zusammen. Mit dem Ziel das Training des Generators auf einen stabilisierten Zustand zu beschleunigen, erhält der Generator aggregiertes Feedback mehrerer Diskriminatoren (vgl. GMAN [21]).

Kombinierte Architektur Eine Kombination aus einer Encoder-Decoder und GAN Architektur kann ebenfalls den Generierungsprozess verbessern. Bekannte Beispiele sind die Modelle ALIGAN [20] und BiGAN [19]. Diese setzen sich aus dem bekannten Generator und Diskriminator sowie einem Encoder Netz zusammen. Während der Generator wie gewöhnlich unter Einfluss des Rauschvektors z neue Daten generiert, erzeugt der Encoder aus den realen Daten Vektoren im latenten Raum. Der Diskriminator wird anschließend nicht nur darauf trainiert reale und generierte Daten zu unterscheiden, sondern lernt auch die Zusammenhänge zwischen Daten im ursprünglichen Raum und der Darstellungen im latenten Raum. Als Eingabe erhält er dafür zusätzlich zu den realen und generierten Daten ihre jeweiligen Vektoren. Im Backpropagation Schritt kann der Diskriminator dem Generator dadurch ein erweitertes Feedback zur Kopplung zwischen dem latenten Raum und den generierten Daten geben.

Verbesserter Diskriminator Diese Architekturvariante konzentriert sich auf ein stabileres Trainingsverhalten des Diskriminators. Das EBGAN [95] zum Beispiel verwendet anstelle eines klassischen Diskriminators eine Auto-Encoder-Architektur. Der Diskriminator erfüllt nicht mehr die Aufgabe eines simplen binären Klassifikators, sondern bewertet mit Hilfe einer Energiefunktion die Qualität der Daten. Die Bewertung zeigt wie realistisch die generierten Daten sind, wobei niedrige Werte auf hochwertige reale Daten und hohe Werte auf generierte minderwertige Daten hinweisen. Wie gängig versucht der Generator den Diskriminator zu täuschen und zielt entsprechend auf niedrige Bewertungen ab.

Netzwerkspeicher Kim et. al [49] erweitern die Architektur um einen GAN-Speicher. Dieser hilft dem Generator verschiedene Klassen sowie Strukturen in den Daten zu unterscheiden und ermöglicht dem Diskriminator auf frühere erzeugte Daten zurückzugreifen. Dadurch wird die Trainingsstabilität des Diskriminators verbessert und der Generator bei der Generierung unterschiedlicher Dateneigenschaften unterstützt.

Flexibler latenter Raum GANs dieser Art erlernen eine verbesserte Rauschverteilung, um unausgewogene Klassenverteilungen innerhalb der Daten auch bei einem geringen Vorkommen berücksichtigen zu können. Das DeLiGAN [38] stellt den latenten Raum als Gaussians Mixture Model (GMM) dar. Während des Modelltrainings werden die Parameter des Mixture Modells optimiert.

2.2.3 Herausforderungen beim Training von GANs

Auch wenn GANs eine hohe Performance erzielen können und zu den fortgeschrittenen Generativen Modellen zählen, bestehen verschiedene Herausforderungen beim Training von GANs. Im Folgenden werden zunächst allgemeine Probleme erläutert und anschließend konkrete Schwierigkeiten im Zusammengang mit tabellarischen Daten nahegelegt.

Allgemeine Herausforderungen

Jabbar et. al [45] geben einen Überblick über den aktuellen Stand von GANs und benennen die wesentlichen Schwierigkeiten beim Training.

Für ein stabiles Training ist es entscheidend ein **Nash-Gleichgewicht** zu erreichen. Beim Nash-Gleichgewicht handelt es sich um ein Konzept aus der Spieltheorie, wobei

die einzelnen Strategien der Spieler voneinander abhängen. In Bezug auf GANs wird entsprechend ein Zustand angestrebt, in dem weder der Generator noch der Diskriminator sich eigenständig optimieren können. Werden unabhängig voneinander Veränderungen angestrebt, kann das Training instabil verlaufen und das Modell evtl. nicht konvergieren. Die Ursache dieses Verhaltens liegt darin, dass sich Gewichtsoptimierungen auf ein Netz positiv und gleichzeitig negativ auf das andere Netz auswirken können.

Ein **Internal Covariate Shift** (ICS) entsteht, wenn Parameteränderungen während des Trainings die Verteilungen von Ausgabedaten der Hidden Neuronen beeinflussen. Schwanken diese Verschiebungen stark, kann dies ebenfalls zu einer erschwerten Konvergenz führen. Um dieser Problematik entgegenzuwirken, muss die Lernrate minimiert werden, was zu einer erhöhten Trainingszeit des Modells sowie einem höheren Ressourcenverbrauch führt.

Eine der am häufigsten genannten Herausforderungen bildet der **Mode Collapse** [67, 17, 40, 45]. Der Generator konzentriert sich ausschließlich auf die Generierung ähnlicher Klassen der abzubildenden Datenverteilung. Die generierten Daten besitzen eine geringe Datenvielfalt und berücksichtigen ausschließlich einen Teil des gesamten Datenbestands.

Eine weitere Herausforderung ist das Problem des **verschwindenden Gradienten** (engl. Vanishing Gradient). Wenn der Diskriminator schnell konvergiert und er die generierten Daten von den realen unterscheiden kann, wird der an den Generator zurückgegebene Gradient sehr klein. Der Generator kann seine Gewichte höchstens minimal optimieren. Verstärkt werden kann die Abnahme des Gradienten zudem durch die Verwendung bestimmter Aktivierungsfunktionen (bspw. Sigmoid) sowie einer großen Anzahl von Schichten. Der Gradient nimmt durch die Multiplikation der Ableitungen exponentiell ab. Das globale Optimum wird nicht erreicht.

Aufgrund des Min-Max Spiels zwischen Generator und Diskriminator ist die Evaluation eines GANs besonders komplex und zeitaufwändig. Kostenfunktionen wie bei der Bewertung einfacher Neuronaler Netze sind nicht nutzbar. Es fehlen **geeignete Bewertungsmetriken**, um die Performance sowie die Trainingsstabilität unterschiedlicher GANs evaluieren und vergleichen zu können. In Abhängigkeit des Datentyps, der Domäne und Motivation der Datensynthese ist die Wahl der Metriken einzeln zu treffen [67].

Herausforderungen mit tabellarischen Daten

Zusätzlich zu den allgemeinen Schwierigkeiten beim Training von GANs sind die komplexen Eigenschaften tabellarischer Daten im Rahmen der Herausforderungen zu berücksichtigen [87]:

Gemischte Datentypen In der Regel bestehen tabellarische Daten aus unterschiedlichen Datentypen. Im Wesentlichen wird zwischen diskreten und kontinuierlichen Typen unterschieden. Im Gegensatz zu diskreten Daten, die eine endliche Anzahl von möglichen Werten besitzen, haben die kontinuierlichen Daten einen unendlichen Wertebereich. Die Aufbereitung und Verarbeitung der Daten unterscheiden sich nach Datentyp.

Nicht-Gaußsche Verteilungen Dadurch, dass die kontinuierlichen Datenwerte für gewöhnlich keiner Gauß-ähnlichen Verteilung entsprechen, führt eine Normalisierung nach einer Min-Max Transformation zu verschwindenden Gradienten.

Multimodale Verteilungen Unter einem Modus wird in der Wahrscheinlichkeitstheorie ein Wert verstanden, der besonders häufig in einem Datensatz vorkommt. Kontinuierliche Spalten bestehen zu meist aus komplexen Datenverteilungen, die sich aus mehreren Modi zusammensetzen. GANs zeigen Schwierigkeiten bei der Modellierung dieser multimodalen Verteilungen [69].

One-hot-encoded Vektoren Die diskreten Trainingsdaten werden als One-hot-encoded Vektor dem Diskriminator zur Verfügung gestellt. Die vom Generator erzeugten diskreten Daten bestehen jedoch nicht aus Vektoren mit einer eindeutigen Zuordnung, sondern beinhalten Wahrscheinlichkeiten zur Zugehörigkeit zu einzelnen Kategorien. Der Diskriminator kann die realen und generierten Daten einzig an ihrem Format identifizieren.

Unausgewogene kategoriale Spalten Viele diskrete Datensätze sind stark unausgewogen und besitzen eine Hauptkategorie, die mehr als 90% der Daten ausmacht. Nebenkategorien sind aufgrund der geringen Menge an Trainingsdaten zum einen schwer zu erlernen, zum anderen verursacht das Fehlen von Nebenklassen nur geringe Änderungen in der Datenverteilung. Diese sind schwer für den Diskriminator zu erkennen und fördern die ausschließliche Generierung der Hauptkategorien.

2.2.4 Datenvorverarbeitung

Infolge der Herausforderungen bei der Verwendung tabellarischer Daten entwickeln Xu et al. [87] grundlegende Verfahren zur Datenvorverarbeitung, die auch in aktuellen Modellen integriert sind (vgl. [97, 77]). Um unterschiedliche Methoden je nach Datentyp anwenden zu können, werden die Datenspalten als diskret oder kontinuierlich gekennzeichnet.

Da die **kontinuierlichen Daten** häufig nicht gaußförmige, aber multimodale Datenverteilungen aufweisen, wird für diese ein **Mode-Specific Normalization (MSN)** [87] Verfahren vorgeschlagen. Mittels eines Variational Gaussian Mixture Models (VGM) wird die Anzahl von Modi innerhalb der Datenverteilung geschätzt. Anschließend wird für jeden ermittelten Modus eine Hilfsspalte angelegt und für jeden Wert der ursprünglichen kontinuierlichen Spalte der am besten passende Modus ermittelt. Die Speicherung folgt einer One-Hot-Kodierung, wobei die Spalte des zutreffenden Modus mit einer Eins und alle restlichen mit Nullen versehen werden. Abschließend wird der ursprüngliche kontinuierliche Wert anhand des ausgewählten Modus normalisiert.

Die Verarbeitung der **diskreten Daten** beschränkt sich auf eine One-Hot-Kodierung. Im Gegensatz zu den kontinuierlichen Werten, die nach der Vorverarbeitung wieder in einer Spalte gespeichert werden, erhöhen die diskreten Daten die Anzahl an Dimensionen der tabellarischen Daten. Für jede Kategorie der diskreten Datenspalte wird eine neue Spalte angelegt. Das Problem von unausgewogenen kategorialen Spalten wird unter Inanspruchnahme eines für den Generator vorgeschalteten Vektors (Conditional Vektor) verbessert. Mit dem **Training-by-Sampling** [87] Ansatz lernt das Modell nicht mehr die gesamte Datenverteilung, sondern eine Verteilung in Abhängigkeit zur ausgewählten Kategorie einer diskreten Spalte.

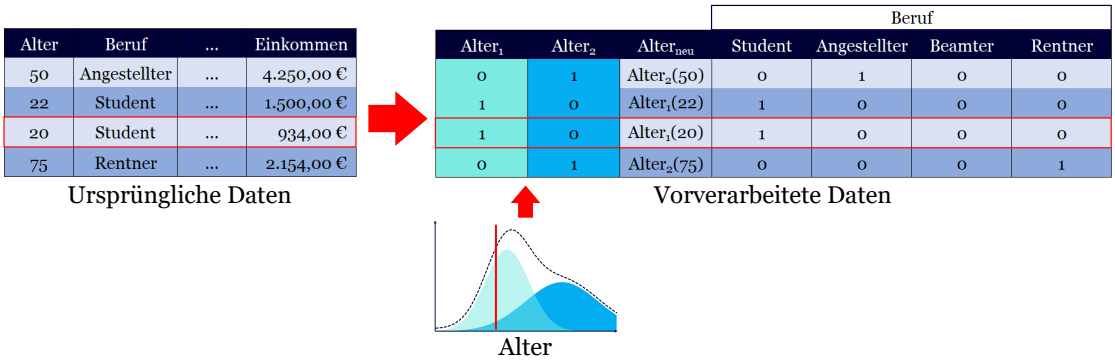

Abbildung 2.3: Verarbeitung kontinuierlicher und diskreter Daten (eigene Darstellung)

Abbildung 2.3 zeigt beispielhaft die Verarbeitung der beiden Datentypen. Die Spalte „Alter“ repräsentiert den kontinuierlichen Datentyp. Mit dem VGM werden zwei Modi identifiziert, die Alters-Werte einem Modus zugeordnet und im Anschluss anhand des Modus Mittelwert und Standardabweichung das Alter normalisiert. In der Spalte $Alter_{neu}$ befinden sich die normalisierten Werte zwischen null und eins. Die One-Hot-Kodierung von diskreten Datentypen wird anhand der Spalte „Beruf“ verdeutlicht.

2.2.5 Evaluation tabellarischer Daten

Da GANs zu dem unüberwachten Lernansatz zählen und das Min-Max-Spiel zwischen Generator und Diskriminator nicht direkt interpretierbar ist, sind Modell-Genauigkeit und Fehlerrate nicht in der Art messbar, wie es bei überwachten Lernansätzen mit Minimierungsproblem der Fall ist [40]. Aus diesem Grund wird die Leistung der GANs häufig ausschließlich anhand der Vielfalt und Qualität der generierten Daten gemessen. Es fehlen einheitliche Metriken, um verschiedene GAN-Algorithmen direkt miteinander vergleichen zu können und die einzelnen Modelle präziser zu optimieren. Je nach Anwendung werden aktuell unterschiedliche Datenevaluationen vorgeschlagen und verwendet. Die folgenden Verfahren konzentrieren sich auf die Evaluation tabellarischer Daten.

Expertenmeinung Um insbesondere die Logik in den Daten stichpunktartig zu überprüfen, können Personen mit Fachwissen die Daten subjektiv bewerten. Beispielsweise sind Bewertungen zur Realitätsnähe und Nützlichkeit der Daten denkbar [7].

Visualisierung Die Exploration der generierten Daten kann durch grafische Methoden vertieft werden. Für einzelne Spalten eigenen sich z.B. die Abbildung von Wahrscheinlichkeitsverteilungen, Histogrammen oder Box-Plots. Ausreißer und die Vielfalt der Datenmodi können sichtbar gemacht werden. Mit der bivariaten Analyse lassen sich die Beziehungen zwischen Features abbilden. Streudiagramme, Heatmaps und Korrelationsdiagramme sind hierfür gängige Visualisierungen.

Statistische Metrik Die Berechnung statistischer Eigenschaften hilft beim Vergleich von generierten und realen Daten. Metriken wie der Durchschnitt, Standardabweichung, Minima und Maxima oder der Median können erste Hinweise auf Abweichungen und Gemeinsamkeiten zwischen generierten und realen Daten geben [25].

Distanzmetrik Für die direkte Berechnung der statistischen Ähnlichkeit von realen und generierten Daten bestehen konkrete Metriken. Die Jensen-Shannon-Divergenz (JSD) und die Wasserstein Metrik berechnen die Differenz der Wahrscheinlichkeitsmassenverteilungen einzelner diskreter Spalten. Darüber hinaus kann die Korrelation zwischen zwei Features durch den Korrelationskoeffizienten nach Pearson (kontinuierliche Daten) oder Unsicherheits-Koeffizienten nach Theil (diskrete Daten) bestimmt werden. Die Berechnung überprüft, ob Wechselwirkungen zwischen den Merkmalen in den generierten Datensätzen erhalten bleiben. Beim anschließenden Vergleich der Korrelationsmatrizen von realen und generierten Daten werden Unterschiede bzw. Gemeinsamkeiten sichtbar [96].

Datenschutz-Metrik Weitere Metriken zur Distanz zwischen den Datensätzen geben Hinweise auf den Schutz der Privatsphäre. Beim Distance to Closest Record (DCR) und Nearest Neighbour Distance Ration (NNDR) wird der euklidische Abstand zwischen einem generierten Datenpunkt und seinen nächsten realen Nachbarn gemessen. Je größer die Werte, desto höher auch der Datenschutz [96].

Klassifikator Test Eine zusätzliche Möglichkeit die Ähnlichkeit der generierten sowie realen Daten zu bewerten, bietet ein Klassifikator. Generierte und reale Daten werden gelabelt und in Trainings- sowie Testdaten eingeteilt. Der Klassifikator lernt anhand der Trainingsdaten die Daten in real und generiert zu unterteilen. Kann das trainierte Modell anschließend die Testdaten unterscheiden, sind die Unterschiede zwischen realen und generierten Daten groß.

ML-Modell Dieses Evaluationsverfahren prüft, ob sich die generierten Daten für das Training von ML-Modellen genauso eignen wie die realen Daten. Die realen Daten werden in Trainings- sowie Testdaten unterteilt und das GAN mit den Trainingsdaten trainiert. Anschließend wird ein zu testendes ML-Modell (z.B. Entscheidungsbaum oder logistische Regression) mit den Trainingsdaten ($Modell_{Real}$) sowie ein zweites mit den generierten Daten ($Modell_{Fake}$) trainiert. Anschließend werden beide Modelle mittels der realen Testdaten bewertet. Mithilfe von weiteren Metriken wie der Genauigkeit oder des F1-Scores können schließlich die Leistungen der ML-Modelle verglichen werden [25, 96].

2.3 Theorie der Differential Privacy (DP)

Über die letzten Jahre waren Angriffe auf ML-Modelle erfolgreich bei der Identifizierung zugrunde liegender Daten [88, 61]. Da diese Modelle häufig sensible Daten aus unterschiedlichen Bereichen wie z.B. Gesundheitswesen, Finanzwesen oder Verkehrswesen verarbeiten, ist der Schutz der Privatsphäre ein entscheidender Bestandteil bei ihrer Entwicklung. Darüber hinaus fordern die unterschiedlichen Datenschutzvorschriften wie die Datenschutzgrundverordnung der EU (DSGVO) oder die „Federal Policy for the Protection of Human Subjects“ der USA Daten zu schützen. Nach der Vorstellung einzelner traditioneller Techniken zum Schutz von Daten wird das für die synthetische Datenverarbeitung zahlreich verwendete mathematische Konzept Differential Privacy erläutert [65].

2.3.1 Traditionelle Anonymisierungsverfahren

Zwei verbreitete Ansätze zum Schutz der Privatsphäre sind die k-Anonymität [78] sowie ihre Erweiterung l-Diversität [55]. Beide Verfahren teilen ein Datenmodell bestehend aus Identifikatoren, Quasi-Identifikatoren sowie sensiblen Attributen. Als **Identifikatoren** werden die Datenpunkte bezeichnet, die zur eindeutigen Identifizierung einer Person führen können. Beispiele sind der Name oder eine persönliche ID. **Quasi-Identifikatoren** ermöglichen in Kombination mit weiteren Quasi-Identifikatoren Rückschlüsse auf eine Person, wie z.B. Postleitzahl, Alter oder Geschlecht. Datenpunkte, die sensible Informationen über eine Person enthalten (z.B. eine explizite Krankheit), zählen zu den **sensiblen Attributen**.

Sweeney [78] schlägt die **k-Anonymität** zum Datenschutz vor. Bei dieser Methodik werden im ersten Schritt alle Identifikatoren aus dem Datensatz entfernt und anschließend die verbliebenen Daten in k-Gruppen bzw. Äquivalenzklassen unterteilt. Die Gruppierung ergibt sich aus den Einträgen, die dieselben Quasi-Identifikatoren besitzen. Die anschließende Generalisierung der (quasi-)identifizierenden Attribute, wie z.B. Alterspannen, verhindert Rückschlüsse auf Verbindungen zwischen sensiblen Attributen und einzelnen Personen. Da bei der k-Anonymität die Privatsphäre einer Person jedoch durch verschiedene Angriffe erheblich verletzt werden kann, wurde die **l-Diversität** [55] entwickelt. Aufbauend auf der k-Anonymität ermöglicht sie eine höhere Datenschutzgarantie,

indem sie mindestens ein unterschiedlich sensibles Attribut je Äquivalenzklasse voraussetzt. Auch wenn die l-Diversität im Vergleich zur k-Anonymität Daten besser schützen kann, besteht weiterhin die Gefahr des Verlustes von hochsensiblen Informationen.

2.3.2 (ϵ, δ)-Differential Privacy

Mit dem Ziel eine Datenschutzgarantie zu realisieren, die auch bei jeglichem Hintergrundwissen eingehalten wird, definiert Dwork [22] das mathematische Konzept Differential Privacy (DP). Im Wesentlichen werden die Informationen geschützt, indem Unterschiede zwischen verschiedenen Datensätzen verborgen bleiben. Dadurch soll verhindert werden, dass zu viele Informationen über eine bestimmte Person ermittelt werden können, ohne allgemeine Muster innerhalb der Datenbasis zu verlieren. Die Grenzen an zur Verfügung stehenden Informationen werden durch die Parameter Epsilon (ϵ) und Delta (δ) festgelegt und mittels eines randomisierten Algorithmus M auf einen bestimmten Datensatz D angewendet. Beispiele für randomisierte Algorithmen sind das Training von GANs oder sonstige ML-Modelle.

Das **Privatsphären Budget** ϵ definiert den maximal gestatteten Verlust der Privatsphäre. Konkret wird die maximale Differenz zwischen Analyseergebnissen benachbarter Datensätze festgelegt. Unter benachbarten Datensätzen werden zwei beliebige Datensätze verstanden, die sich genau in einem Datenpunkt unterscheiden. Je kleiner der ϵ-Wert, desto besser sind die Daten geschützt. Der zweite begrenzende Parameter ist die **Fehlerwahrscheinlichkeit** δ. Sie gibt die Wahrscheinlichkeit für einen Verstoß gegen DP an und sollte dementsprechend einen sehr niedrigen Wert besitzen. Zusammenfassend lässt sich Differential Privacy wie folgt definieren.

Definition 2.1 (ϵ, δ)-Differential Privacy) *Sei $M : D \to R$ ein randomisierter Algorithmus. M erfüllt (ϵ, δ)-DP mit $\epsilon \in \mathbb{R}^+$ und $\delta \in [0, 1]$, falls für alle benachbarten Datensätze D und D' sowie für alle möglichen Teilmengen der Ausgabemenge $S \subseteq R$ gilt, dass*

$$Pr[M(D) \in S] \leq e^{\epsilon} \cdot Pr[M(D') \in S] + \delta \tag{2.4}$$

In Worten beschrieben besagt die Definition, dass sich die Wahrscheinlichkeiten der Produktion eines Ausgabewertes in der Menge S zwischen den Mechanismen M(D) und

M(D') nicht mehr unterscheiden dürfen als es die Multiplikation mit e^{ϵ} und Addition mit δ zulassen. Eine Erweiterung zum einfachen DP, bietet die **Rényi Differential Privacy (RDP)** [58]. Sie verwendet im Vergleich zum einfachen DP strengere Grenzen für das Privatsphären Budget ϵ und bietet Vorteile insbesondere in Bezug auf das Training von GANs sowie bei der Verarbeitung großer Datenräume. RDP basiert auf der Rényi Divergenz, der die Ähnlichkeit zwischen Verteilungen berechnet.

Um Differential Privacy einzuhalten und die Ergebnisse der benachbarten Datensätze anzugleichen, wird mathematisches Rauschen verwendet. Aus einer statistischen Verteilung wie z.B. Gauß- oder Laplace-Verteilung wird ein zufälliger Rauschvektor gebildet und auf die realen Ergebnisse addiert. Die wahren Ergebnisse können nach der Addition nicht mehr mit Sicherheit prognostiziert werden. Die Verteilung steht im Zusammenhang mit dem gewählten Privatsphären Budget ϵ. Ein niedrig gewählter ϵ-Wert führt zu einem hohen Rauschen und folglich zu einem erhöhten Datenschutz. Die Kalibrierung des Rauschens auf den definierten ϵ-Wert wird durch die Berechnung der Parameterwerte des Rauschmechanismus anhand der Sensitivität ermöglicht. Diese beschreibt den maximalen Abstand, um den sich der Output zwischen den benachbarten Datensätzen verändern darf [10].

2.3.3 Local vs. Global Differential Privacy

Die Umsetzung von Differential Privacy unterscheidet sich in den Ansätzen **Local Differential Privacy (LDP)** und **Global Differential Privacy (GDP)**[90, 56]. Das lokale Modell zielt darauf ab, eine Datenbank mit bereits geschützten Daten zu entwickeln. Der randomisierte Algorithmus wird direkt auf die Daten der einzelnen Personen angewendet und anschließend in der Datenbank gespeichert. Bereits während der Erfassung der individuellen Informationen werden die Daten geschützt, nicht erst bei der Abfrage von Daten. Der Vorteil besteht darin, dass Data Scientisten aufgrund des Post-Processing Theorems unzählig viele Abfragen an die Datenbank stellen können, ohne die Differential Privacy Garantie zu verletzen.

Das Post-Processing Theorem besagt, dass die Ergebnisse aller Berechnungen auf einem DP garantierten Output ebenfalls Differential Privacy erfüllen [23].

Definition 2.2 (Post-Processing Theorem) *Sei $M : \mathbb{N}^{|x|} \rightarrow R$ ein randomisierter Algorithmus der (ϵ, δ)-Differential Privacy erfüllt. Sei f: $R \rightarrow R'$ eine beliebige randomisierte Abbildung. Dann garantiert auch $f \circ M : \mathbb{N}^{|x|} \rightarrow R'$ (ϵ, δ)-Differential Privacy.*

Im Gegensatz zum LDP wird beim Global Differential Privacy der randomisierte Algorithmus auf die Antwort einer Datenbankabfrage der Data Scientisten angewendet. Die Datenbank enthält hierbei unbearbeitete individuelle Informationen der einzelnen Personen. Beim GDP ist das Kompositionstheorem der Differential Privacy von den Anwendern zu berücksichtigen. Schließlich dürfen auch verschiedene Datenbankabfragen zusammen den vorgegebenen ϵ-Wert nicht überschreiten.

Das **Kompositionstheorem** zielt darauf ab, die Garantie der Privatsphäre auch über mehrere Anwendungen hinweg aufrechtzuerhalten [23]. Das Theorem berücksichtigt den Gesamtverlust an Privatheit, welches das Privatsphären Budget nicht überschreiten darf. Die Basis Komposition für die (ϵ, δ)-Differential Privacy addiert für alle Durchläufe k die ϵ-Werte und δ-Werte (siehe Formel 2.5). Während die Basis Komposition konservative Obergrenzen der verwendeten ϵ-Werte und δ-Werte berechnet, können fortgeschrittene Kompositionstheoreme genauere Angaben für die verwendeten Parameter berechnen.

$$(\sum_{i=1}^{k} \epsilon_i, \sum_{i=1}^{k} \delta_i) - DP \tag{2.5}$$

Auch wenn die Local Differential Privacy gegenüber der Global Differential Privacy einen stärkeren Privatsphärenschutz bietet, kann aber das erhöhte Rauschen auf die individuellen Datenpunkte zu stark verzerrten Analyseergebnissen führen und somit die Plausibilität von LDP in Frage stellen [56]. Abbildung 2.4 veranschaulicht die unterschiedlichen Herangehensweisen der beiden Differential Privacy Ansätze.

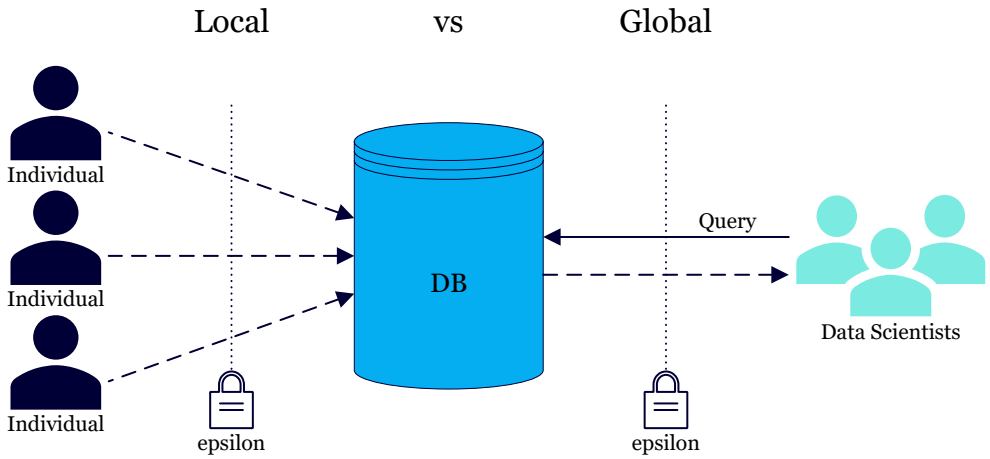

Abbildung 2.4: Local vs. Global Differential Privacy (in Anlehnung an [90])

2.3.4 Trade-Off zwischen Nutzbarkeit und Privatsphäre

Grundsätzlich besteht bei der Arbeit mit sensiblen Daten ein Trade-Off zwischen der Nutzbarkeit der Analysen und der Privatsphäre der Daten [10, 41]. Der Schutz der Privatsphäre erfolgt auf Kosten der Genauigkeit von Analysen. Beim ML führt ein erhöhter Datenschutz zu einer verschlechterten Modellgenauigkeit. Je kleiner das Privatsphären Budget, desto mehr Rauschen muss auf die Ergebnisse addiert werden und desto geringer wird der Analysenutzen. Im Extremfall (ϵ-Wert = 0) muss die Ausgabe immer gleich sein, unabhängig von der Eingabe. In diesem Fall sind Datenanalysen nutzlos. In Abhängigkeit zur Problemstellung muss folglich zwischen der Nutzbarkeit der Analyse und dem Privatsphärenschutz abgewogen werden. Tendenziell wird mit Zunahme der Datenmenge der Trade-Off geringer und es lassen sich auch plausible Analyseergebnisse unter erhöhtem Privatsphärenschutz erzielen.

2.4 Praktiken zur Integration von DP in GANs

Um ML-Modelle unter Datenschutzgarantien zu entwickeln, wird in vielen Ansätzen die Einbettung von Differential Privacy in die Trainingsphase der Modelle empfohlen [2, 92, 75]. Zu den bekanntesten Verfahren der Integration von DP in GANs zählen Differentially Private Stochastic Gradient Descent (DP-SGD) [2] sowie Private Aggregation of Teacher Ensembles (PATE) [63].

2.4.1 Differentially Private Stochastic Gradient Descent (DP-SGD)

Differentially Private Stochastic Gradient Descent [2] ist eine häufig verwendete Methode, um Differential Privacy in Deep-Learning-Modelle zu integrieren. Mit der Grundidee, dass der Einfluss einzelner Trainingspunkte auf die Aktualisierung der Modellparameter begrenzt wird, soll das Modell keine sensiblen Informationen über einzelne Trainingsdaten lernen können. Beim einfachen SGD wird der Gradienten-Vektor für jedes einzelne Trainingsdatum unmittelbar in die Aktualisierung der Modellparameter einbezogen (siehe Kapitel 2.1.2). Ein einzelnes Datum kann dadurch zu starken Veränderungen der Modellparameter führen und das trainierte Modell nachfolgend sensible Informationen einzelner Datensätze preisgeben.

Um das zu verhindern, werden beim DP-SGD die Gradienten der einzelnen Daten modifiziert. Im Detail werden die Gradienten in einem ersten Schritt gekürzt (engl. clipping) und anschließend kalibriertes Rauschen (engl. noise) hinzugefügt. Mittels einer definierten Clipping Grenze kann der Gradient auf einen Bereich begrenzt und die Sensitivität eines Trainingsschritts unter Kontrolle gehalten werden. Die Wahl der richtigen Größe für die Clipping Grenze ist entscheidend. Während ein zu starkes Verkürzen der Gradienten zu einem Verlust von zentralen Informationen führen kann, schützt ein zu niedrig gewählter Grenz-Wert die Privatsphäre nicht ausreichend. Der Trade-Off zwischen Nutzbarkeit und Privatsphärenschutz wird sichtbar. Das im zweiten Schritt hinzugefügte Rauschen folgt einer Gauß-Verteilung mit dem Rauschfaktor σ und bewirkt die zufällige Verschiebung des Gradienten. Informationen über individuelle Datenpunkte werden verfälscht und gemittelt. DP-SGD wird in der Regel ausschließlich auf das Training des Diskriminators angewendet. Aufgrund des Post-Processing Theorems (vgl. Definition 2.2) und der Tatsache, dass der Generator die realen Trainingsdaten nicht verwendet, erfüllt ein auf einem differentiell privaten Diskriminator trainierter Generator ebenfalls Differential Privacy. Abbildung 2.5 zeigt den Aufbau des DPGANs [85], eines der ersten GANs das DP-SGD verwendet.

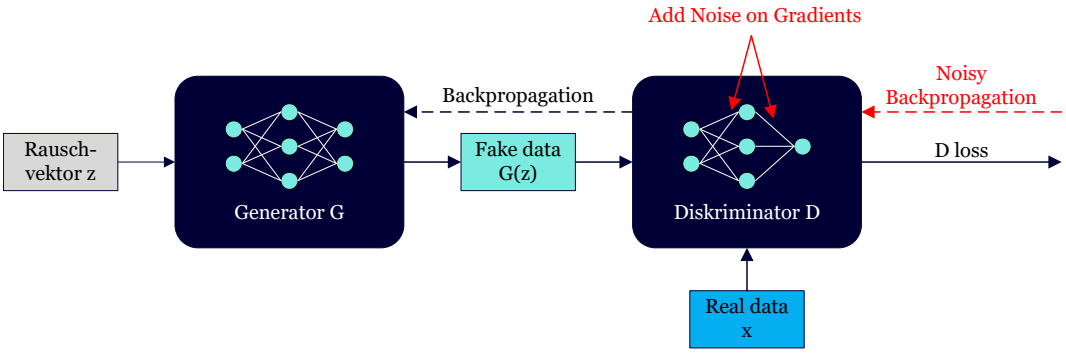

Abbildung 2.5: Aufbau und Trainingsprozess des DPGANs (in Anlehnung an [46])

Um die Trainingsstabilität und Konvergenz-Rate zu verbessern, werden unterschiedliche Optimierungsstrategien zum **Gradienten-Clipping** vorgeschlagen [93]. Beispielsweise passt ein Adaptives Clipping die Clipping-Werte während des Trainings an und eine Parametergruppierung fasst Parameter mit ähnlichen Clipping Grenzen zusammen, um einen Kompromiss zwischen dem Privatsphären-Verlust und der Konvergenz jeder Gruppierung zu erreichen. In [28] wird eine Clipping Decay Strategie vorgestellt. Diese ermöglicht das Rauschen in Abhängigkeit zur Größe der Gradienten zu reduzieren. Der Clipping-Wert selbst nimmt hierbei mit jedem Generator-Update exponentiell ab. Unter Verwendung der **Wasserstein-Kostenfunktion (Wloss)** [4] in Verbindung mit dem **Gradient Penalty Term** [37] kann der Clipping-Wert zudem automatisch erzwungen werden. Ein zu starkes Beschneiden der Gradienten wird beim Gradient Penalty durch das Einhalten der Lipschitz-Kontinuität verhindert. Ein explizites Clipping der Gewichte wird nicht mehr erforderlich, was zu einer erhöhten Trainingsstabilität führt.

Das auf der Wloss aufbauende **Wasserstein-GAN (WGAN)** [4] versucht im Wesentlichen das Problem des Mode Collapse zu reduzieren. Im Gegensatz zum ursprünglichen GAN [34], das die Jensen-Shannon-Divergenz als Maß für die Abstände zwischen den synthetischen und realen Datenverteilungen nutzt, erzielt das fortgeschrittene WGAN eine erhöhte Trainingsstabilität durch die Verwendung der Earth-Mover-Distanz (bzw. Wasserstein-1-Distanz). Die Earth-Mover-Distanz misst die minimalen Kosten für die Transformation von Datenpunkten einer beliebigen Verteilung in eine Zielverteilung. Mit der Absicht real aussehende Daten zu erzeugen, die den Diskriminator täuschen können, wird beim Training des WGANs der Wasserstein-Abstand zwischen der generierten Datenverteilung p_z und der realen Datenverteilung p_{data} minimiert.

Die Anzahl der Trainingsiterationen des Diskriminators wird durch das Privatsphären Budget begrenzt. Auf Basis des Kompositionstheorems (vgl. Formel 2.5) werden die Verluste an Privatsphäre bei jeder Ausführung von differentiell privaten Mechanismen akkumuliert. Um das definierte Privatsphären Budget dennoch nicht zu übersteigen, wird bei jeder Trainingsiteration der Wert mittels des **Privacy Accountant Konzepts** überprüft. Sobald das Gesamtbudget den definierten Wert überschreitet, wird der Trainingsprozess beendet. Zwei weit verbreitete Privacy Accountant Techniken sind **Moment Accountant** [2] und **Rényi Differential Privacy Accountant** [58]. Die Techniken können nicht den exakten Privatsphären-Verlust ermitteln, sondern berechnen den höchstmöglichen Verlust. Da das RDP-Accountant im Vergleich zum Moment Accountant jedoch eine engere Schranke liefert, kann das Modell bei gleich definierten Privatsphären Budget länger trainiert werden und zufolge verbesserte Daten generieren [26].

Ergänzend zum Post-Processing und Kompositionstheorem beschäftigen sich Wang et. al [82] mit den Datenschutzgarantien von Stichproben. Ihr Theorem „**RDP für Subsampled Mechanismen**“ besagt, dass ein Mechanismus M, der RDP erfüllt, den gleichen Schutz garantieren kann, wenn er auf einer Teilmenge der Daten angewendet wird. Durch die Verwendung einer Subsampling-Rate, die sich aus der Division von Batch-Größe durch Datensatzgröße berechnet, entsteht eine weitere Zufälligkeit, die den Datenschutz des Diskriminators zusätzlich verstärkt. Die Wahrscheinlichkeit der Preisgabe von Informationen einzelner Individuen verringert sich auf die Trainingsiterationen, in denen sie vertreten sind. Darüber hinaus führt das Theorem dazu, dass das Privatsphären Budget aufgrund des kleineren Trainingsdatensatzes effizienter und exakter berechnet werden kann.

2.4.2 Private Aggregation of Teacher Ensembles (PATE)

Eine alternative Variante zur Integration von DP in Deep-Learning Modelle bietet das Private Aggregation of Teacher Ensembles Framework [63]. PATE besteht aus einem Ensemble von Lehrermodellen (Teacher) sowie einem Schülermodell (Student). Das Training der Lehrermodelle basiert auf disjunkten Partitionen der sensiblen Trainingsdaten, wobei jedem Lehrermodell eine feste Partition zugeteilt ist. Nach dem Training der Lehrermodelle sind diese in der Lage Vorhersagen für neue Datensätze zu treffen. Aufgrund der unterschiedlichen Trainingsdaten können die Vorhersagen jedoch unterschiedlich ausfallen. Entsprechend werden die Ergebnisse aggregiert und mit Rauschen versehen, um DP zu gewährleisten. Im Anschluss wird das Schülermodell anhand der aggregierten Vorhersagen der Lehrer trainiert und steht schließlich für die Klassifizierung bereit. Dadurch

dass das Schülermodell keinen Zugriff auf die sensiblen Daten besitzt, erfüllen das Schülermodell selbst sowie seine generierten Daten Differential Privacy.

Beim vorgeschlagenen PATE-GAN [46] wird der Diskriminator durch den PATE Mechanismus ersetzt. Der Generator bleibt im Vergleich zum traditionellen GAN unverändert. Sowohl die Lehrermodelle als auch das Schülermodell haben die Aufgabe die eingehenden Daten als real bzw. fake zu identifizieren. Das Training der Lehrermodelle gleicht der Weise des klassischen Diskriminators, jedoch auf Basis einzelner Partitionen. Die größere Neuerung des PATE-GANs ergibt sich durch die Implementierung des Schüler-Diskriminators. Für das Training des Schülermodells klassifizieren die trainierten Lehrermodelle die generierten Daten als real (falsche Prognose) oder fake (richtige Prognose). Darauf werden die einzelnen Prognosen der Lehrer aggregiert und mit Rauschen verfälscht. Im letzten Schritt erfolgt das eigentliche Training des Schülermodells. Anhand der generierten Daten sowie der Prognosen der Lehrer lernt das Schülermodell die generierten Daten zu klassifizieren.

Alle drei Modelltypen (Lehrer, Schüler, Generator) werden iterativ trainiert. Jede Trainingsiteration des Generators besteht aus n_t Aktualisierungen aller Lehrer sowie n_s Aktualisierungen des Schülers. Auch bei dieser Variante ist die Anzahl an Iterationen abhängig vom Privatsphären Budget ϵ. Mit Hilfe des Moment Accountants wird dieses während des Trainings berechnet. Abbildung 2.6 visualisiert den Trainingsprozess des PATE-GANs.

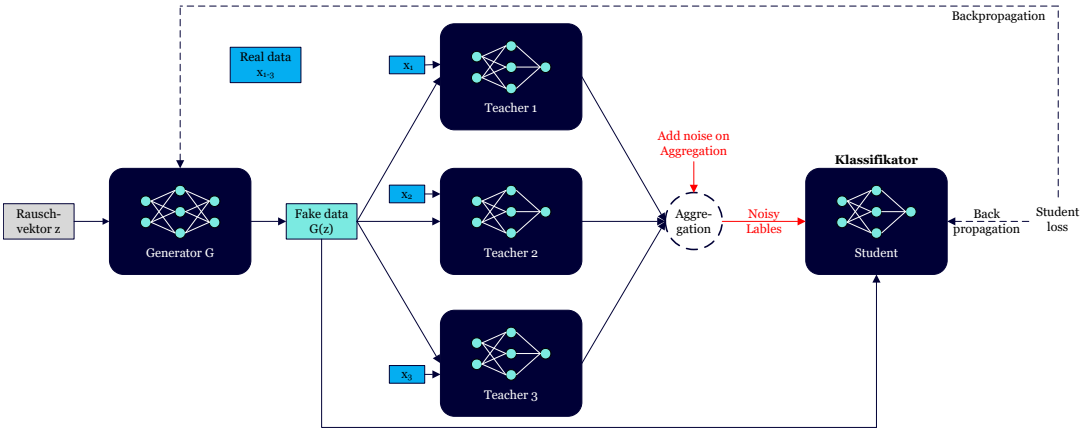

Abbildung 2.6: Aufbau und Trainingsprozess des PATE-GANs (in Anlehnung an [46])

3 Verwandte Literatur

In zahlreichen Arbeiten der letzten Jahre wird Differential Privacy auf das Training von GANs angewendet. Der nachfolgende Überblick vorhandener Modelle und verwandter Literatur beschränkt sich auf GANs, die zum einen den Schutz von Privatsphäre berücksichtigen und zum anderen tabellarische Daten generieren. Zunächst werden die im Hauptteil verwendeten Modelle beschrieben, bevor anschließend weitere Modellarten und Techniken vorgestellt werden.

3.1 Verwendete Modelle

DPGAN Xie et. al [85] entwickeln eines der ersten differential private GANs namens DPGAN, das zusätzlich zur Generierung von tabellarischen Daten auch zur Synthese von Bilddateien verwendet werden kann. DPGAN nutzt für die Integration von DP beim Training des Diskriminators die DP-SGD Methode. Der Generator profitiert von dem Post-Processing Theorem der DP und kann daher auch ohne explizite Gradienten-Verzerrung Differential Privacy garantieren. Verschiedene Arten des Gradienten-Clippings werden ausschließlich auf die Parameter der Gewichte angewendet und durch einen definierten Grenz-Wert beschränkt. Zudem wird für eine verbesserte Trainingsstabilität die Kostenfunktion des WGANs [4] gewählt und mit dem Moment Accountant das Privatsphären Budget kontrolliert.

Die Architektur des DPGANs baut auf der Struktur des Deep Convolutional Generative Adversarial Network (DCGANs) [64] auf und verwendet die Leaky ReLu Aktivierungsfunktion im Diskriminator sowie die ReLu Funktion im Generator. Um ein Overfitting des Modells zu vermeiden, integrieren die Autoren eine L2-Regularisierung in die Aktualisierung der Gewichte beider Netze und führen mit dem Optimierungsalgorithmus RMSProp eine an die Größe der Gradienten sich anpassende Lernrate ein. Für nachfolgende Publikationen bildet DPGAN in vielen Fällen die Basis und wird nahezu immer als Benchmark verwendet.

CTAB-GAN+ Die Autoren des CTAB-GANs [96] konzentrieren sich auf die Generierung tabellarischer Daten. Im Fokus stehen die Verarbeitung von gemischten Datentypen, bestehend aus kontinuierlichen sowie diskreten Datenwerten, und der Umgang mit Datenverteilungen, die lange Verteilungsenden besitzen. Zudem wird die Generierung von verzerrten kontinuierlichen Variablen mit mehreren Modi verbessert. Realisiert werden die dem CTAB-GAN zugrundeliegenden Ziele durch die Erweiterungen der herkömmlichen GAN-Architektur.

Aufbauend auf dem Conditional GAN (CGAN) [59] wird ein Conditional Vektor in Verbindung mit der Training-by-Sampling Methode eingeführt. Diese Techniken ermöglichen Modi gezielt zu erzeugen und einen Mode Collapse zu verhindern. Des Weiteren wird in Ergänzung zum Generator und Diskriminator eine dritte Komponente C als Klassifikator oder Regressor eingeführt. Diesbezüglich wird im Vorfeld eine Spalte des Trainingsdatensatzes ausgewählt, die C anhand der übrigen Zeileninformationen prognostizieren soll. Die Prognose von C hilft dem Generator die semantische Integrität der generierten Daten zu verbessern. Um eine geeignete Darstellung von gemischten Datentypen sowie einen verbesserten Umgang mit fehlenden Daten zu realisieren, wird zudem ein neuartiger Mixed-Type Encoder vorgestellt.

Die Integration von Differential Privacy erfolgt in dem erweiterten Modell CTAB-GAN+ [97]. Zusätzlich zu den Bestandteilen und Zielen des CTAB-GANs wird die Kostenfunktion Wloss in Zusammenhang mit dem Gradient Penality eingeführt und die DP-SGD Methode für das Training des Diskriminators verwendet. Die Gewährleistung für das gewählte Privatsphären Budget wird durch das RDP-Accountant kontrolliert. Abbildung 3.1 visualisiert den Aufbau des CTAB-GAN+.

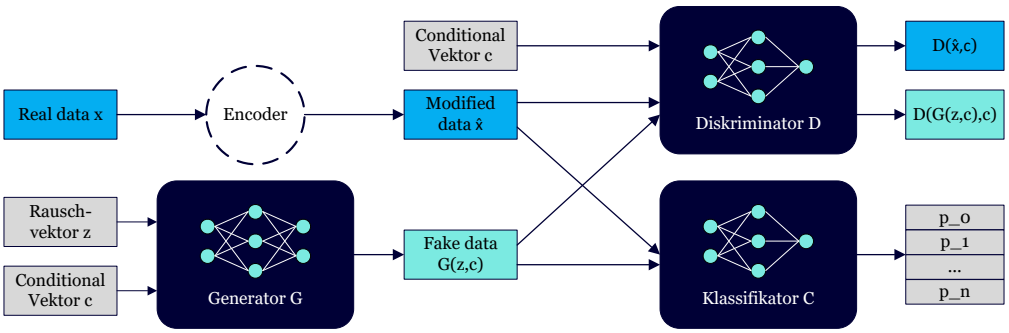

Abbildung 3.1: Aufbau des CTAB-GAN+ (in Anlehnung an [97])

DP-CGANS Genau wie das CTAB-GAN+ nutzt DP-CGANS [77] aufgrund von gruppierten Samples Wloss in Verbindung mit dem Gradient Penality die DP-SGD Methode für das Training des Diskriminators sowie das RDP-Accountant zur Kontrolle des Privatsphären-Verlusts. Auch die Grundstruktur des CGANs wird als Basis implementiert, um mithilfe des Conditional Vektors unterrepräsentierte Klassen nicht zu vernachlässigen. Das Neuartige des DP-CGANS besteht in der ergänzenden Unterstützung bei der Simulation von Korrelationen und Abhängigkeiten zwischen unausgewogenen Variablen. Diese wird ebenfalls mit dem Conditional Vektor umgesetzt. Im Gegensatz zum einfachen Conditional Vektor, bei dem die Datenverteilung in Abhängigkeit zu einer einzigen ausgewählten Kategorie einer diskreten Spalte gelernt wird, ermöglicht DP-CGANS das Erlernen der Verteilungen anhand von zufällig ausgewählten Kategorie-Paaren. Da mit diesem Konzept die Generierung nicht realistischer Konstellationen - wie z.B. Männer und Gebärmutterhalskrebs - unterbunden werden soll, muss die Zusammensetzung des Paares in den realen Daten vorhanden sein.

PATE-GAN Auch wenn die Mehrheit der GANs mit DP-Garantie auf der DP-SGD Methode beruht, erzielt das in Kapitel 2.4.2 vorgestellte PATE-GAN (siehe Abbildung 2.6) vergleichbare Ergebnisse (siehe [46, 97]) auf Basis einer grundlegend anderen Herangehensweise. Zwar stützt es sich ebenfalls auf das Post-Processing Theorem und wendet DP ausschließlich auf das Training des Diskriminators an, jedoch besteht der Diskriminator aus mehreren Modellen. Ein Ensemble von Lehrer-Diskriminatoren unterrichtet einen Schüler-Diskriminator, der die Klassifikation zwischen generierten und realen Daten vornimmt. G-PATE [54] verwendet ebenfalls die PATE-Variante, um DP bei der Generierung von Bildern sowie tabellarische Daten zu berücksichtigen. Beim G-PATE entspricht das Schülermodell dem Generator. Dieser wird mithilfe der aggregierten und verrauschten Gradienten der Lehrermodelle trainiert. Die Autoren erzielen einen geringeren Nutzenverlust der Trainingsdaten bei gleich bleibendem Privatsphären Budget.

GAN	**DPGAN** [85]	**PATE-GAN** [46]	**CTAB-GAN+** [97]	**DP-CGANS** [77]
Autoren	Xie et. al	Jordon et. al	Zhao et. al	Sun et. al
Jahr	2018	2019	2022	2023
DP Algorithm	DP-SGD	PATE	DP-SGD	DP-SGD
Kostenfunktion	Wloss	Kullback–Leibler Divergence	Wloss & Gradient Penality	Woss & Gradient Penality
Gradient Clipping	Ja	Nein	Nein	Nein
Noise	Gaussian Noise	Laplacian Noise	Gaussian Noise	Gaussian Noise
Accountant	Moment Accountant	Moment Accountant	RDP-Accountant	RDP-Accountant
Grundarchitektur	Deep Convolutional GAN (DCGAN)	-	Conditional GAN (CGAN)	Conditional GAN (CGAN)
Erweiterungen	-	+ mehrere Diskriminator-Modelle (Schüler + Lehrer)	+ Klassifikatior / Regressor Komponente C + Mixed-Type Encoder	+ Conditional Vektor für Kategorie-Paare
Datenart	Tabellarische Daten & Bilder	Tabellarische Daten	Tabellarische Daten	Tabellarische Daten
Domäne	Gesundheitsdaten & Zahlen	Gesundheitsdaten & Verschiedenes	Verschiedenes	Sozioökonomische Daten & Gesundheitsdaten

Tabelle 3.1: Verwendete Modelle sortiert nach Erscheinungsjahr

Zusammenfassend zeigt Tabelle 3.1 die maßgeblichen Unterschiede der verwendeten Modelle.

3.2 Weitere Literatur

Federated Learning Während die meisten Differential Private GANs mit zentralisierten Trainingsdaten arbeiten, wird bei einzelnen Ansätzen Federated Learning berücksichtigt. Beim Federated Learning werden ML-Modelle auf Basis von verteilten Trainingsdaten trainiert. Im Wesentlichen unterscheiden sich der verteilte und zentralisierte Ansatz in dem Speicherort der Trainingsdaten sowie dem Ausführungsort der Modelle. Das **Federated Average GAN (Fed-Avg GAN)** [6] garantiert Differential Privacy auf Benutzerebene. Bei jeder Iteration stellt ein zentraler Server das GAN-Modell für eine Teilmenge der Geräte zu Verfügung. Der Diskriminator wird lokal von den einzelnen Endgeräten mittels derer privater Daten trainiert und die Aktualisierung zurück zum Server gesendet. Der Server aggregiert anschließend alle Änderungen und verfälscht sie mit Rauschen. Ein weiteres bekanntes Modell ist **GS-WGAN** [15]. Im Vergleich zum Fed-Avg GAN werden die Diskriminatoren nicht zwischen dem Server und Endgerät geteilt, sondern auf Letzterem gespeichert. Darüber hinaus werden die neu berechneten Gradienten nicht erst auf dem Server, sondern direkt auf den Endgeräten verfälscht.

Privatsphäre vs. Genauigkeit Ergänzend um die Vorstellung einzelner Differential Private GANs beschäftigen sich einige Arbeiten mit dem Trade-Off zwischen Privatsphäre und Genauigkeit. Kossen et. al [50] entwerfen ein GAN für den Bereich der Neurobildgebung. Innerhalb ihrer Evaluation fokussieren sich die Autoren insbesondere auf den Einfluss unterschiedlicher Privatsphären Budgets. Während sie bei einem ϵ-Wert von 7,4 im Vergleich zum Training ohne Differential Privacy nur geringe Einbußen in der Genauigkeit erreichen, sinkt die Leistung bei ϵ-Werten kleiner fünf so stark, dass die generierten Bilder unbrauchbar sind. Um ein geeignetes Verhältnis zwischen Genauigkeit und Privatsphäre zu erhalten, entwickelt Bernau [9] Techniken zur Quantifizierung eines geeigneten Maßes an Privatsphäre. Diesbezüglich werden zwei Hauptprobleme untersucht. Zum einem wird der minimal notwendige Grenzwert an Privatsphären-Verlust durch Membership Inference-Angriffen gesucht; zum anderen wird überprüft wie die Differential Privacy Garantien mit rechtlichen und ethischen Normen in Verbindung gebracht werden können.

Individualisierte Differential Privacy Mit der Annahme, dass verschiedene Personen auch unterschiedliche Erwartungen in Bezug auf Privatsphäre besitzen, erweitern Boenisch et. al die Methoden PATE [12] sowie DP-SGD [11] um individualisierte Differential Privacy Garantien. Während sich ein einheitliches Privatsphären Budget an der strengsten Datenschutzanforderung aller Datenbesitzer orientieren muss, können bei der Verwendung individualisierter Privatsphären Budgets Datenpunkte mit geringeren Anforderungen verbesserte Informationen für das ML-Training bereitstellen. Infolgedessen kann die Leistung des Modells erhöht werden, ohne einzelne Datenschutzanforderungen zu verletzen.

ADS-GAN Im Gegensatz zu den vorherigen Modellen und Methoden verwendet das ADS-GAN [89] kein Differential Privacy. Stattdessen führen die Autoren die Epsilon-Identifizierbarkeit ein. Ihre Grundannahme besteht darin, synthetische Datenpunkte zu generieren, die sich ausreichend von den realen Datenpunkten unterscheiden. Daher sollte der Abstand jedes Datenpunkts im realen Datensatz zu seinem nächsten synthetischen Datenpunkt kleiner sein als der Abstand zum nächsten realen Datenpunkt. Diese Mindestentfernung wird dabei mithilfe des euklidischen Abstands berechnet. Bis auf die unterschiedliche Herangehensweise der Messung von Privatsphäre unterscheidet sich ADS-GAN wenig von den zuvor vorgestellten DP-GANs. Ebenfalls wird das Wasserstein-GAN mit Gradient Penalty für eine verbesserte Trainingsstabilität eingesetzt sowie das CGAN als Basis für eine ausgewogene Datensynthese verwendet.

4 Experimenteller Aufbau

Im Folgenden werden basierend auf den Anforderungen der DaFne-Plattform die Zielsetzung sowie Forschungsfrage der Thesis definiert. Ferner wird die zentrale Fragestellung untergliedert und ein Überblick über die einzelnen Experimente gegeben. Darüber hinaus werden die Architekturen inklusive Modifikationen der ausgewählten Modelle, Trainingsdatensätze sowie verwendete Evaluations-Metriken erläutert.

4.1 DaFne Plattform

Das der These zugrundeliegende Forschungsprojekt „Data Fusion Generator für die Künstliche Intelligenz“ (DaFne) verfolgt das Ziel tabellarische Daten für KI-Applikationen bereitzustellen. Vom Anwender benötigte Daten lassen sich auf einer frei verfügbaren Plattform auf unterschiedliche Weise generieren. Im Wesentlichen stehen die drei Synthesemethoden Reproduktion, regelbasierte Erzeugung und Daten Fusion zur Verfügung. In Abhängigkeit vom Anwendungsfall, der bereits vorhandenen Daten und Komplexität der benötigten Daten wird eine der drei Methoden ausgewählt.

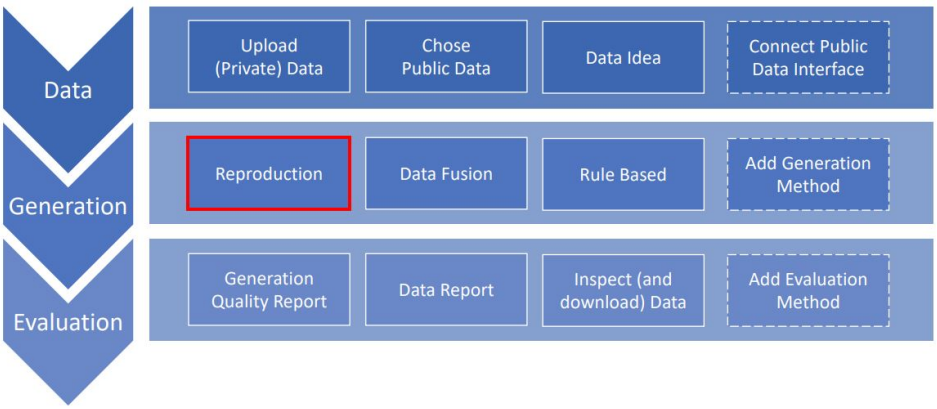

Abbildung 4.1: Überblick über die Funktionalitäten der DaFne-Plattform [51]

Abbildung 4.1 gibt einen Überblick über die grundlegende Funktionalität der Plattform. Konkret zielen die Experimente dieser Arbeit auf eine Erweiterung im Bereich der Reproduktion Methode (rot markiert) ab. Bei der Reproduktion werden auf Basis vorhandener Daten Eigenschaften sowie Verteilungen gelernt und anschließend verwendet, um neue Daten zu generieren. Grundsätzlich zielt die Methode darauf ab, einen kleinen Datensatz um weitere Zeilen zu erweitern, ohne ursprüngliche Dateneigenschaften zu verändern.

In dieser Arbeit wird sie nicht primär für die Vergrößerung eines Datensatzes, sondern vielmehr für die Sicherstellung von Privatsphäre in schützenswerten Daten genutzt. Aus einem vorhandenen Datensatz werden Muster gelernt und nachfolgend ein vollständig neuer Datensatz generiert. Rückschlüsse auf ein reales Datum sind nicht mehr möglich. Die für den privaten Reproduktion-Service zu berücksichtigenden funktionalen und nichtfunktionalen Anforderungen werden folgend definiert.

4.1.1 Funktionale Anforderungen

Die funktionalen Anforderungen lassen sich in die Kategorien Modellaufbau, Modelltraining sowie generierte Daten unterteilen. Aus Sicht des Nutzers ergeben sich die folgenden funktionalen Anforderungen für einen privaten Reproduktion-Service:

Id	Beschreibung
	Modellaufbau
FA_01	Modell garantiert beim Training Privatsphäre
FA_02	Architektur des Modells ist skizziert
FA_03	Modelltraining, Vor- und Nachbearbeitung sind nachvollziehbar
FA_04	Trainiertes Modell bleibt für erneute Datensynthese gespeichert
	Modelltraining
FA_05	Beispieldatensatz steht zur Verfügung
FA_06	Trainingsparameter sind frei wählbar
FA_07	Hilfestellung bei der Wahl der Trainingsparameter ist vorhanden
FA_08	Default Trainingsparameter werden angezeigt
FA_09	Trainingsdauer und benötigte Epochenanzahl werden prognostiziert
FA_10	Trainingsfortschritt der Generierung wird angezeigt
FA_11	Sobald die Daten generiert sind erfolgt eine Benachrichtigung

	Generierte Daten
FA_12	Unabhängig von Eigenschaften und Domäne sind Trainingsdaten wählbar
FA_13	Generierte Daten lassen keinen Rückschluss auf reale Daten zu
FA_14	Generierte Daten besitzen die Eigenschaften der realen Daten
FA_15	Metriken zur Überprüfung der einzuhaltenden Privatsphäre sind verfügbar
FA_16	Metriken zur Qualitätsüberprüfung sind vorhanden
FA_17	Generierte Daten sind in gleicher Weise geeignet für KI-Anwendungen

Tabelle 4.1: Funktionale Anforderungen an einen privaten Reproduktion-Service

4.1.2 Nicht-Funktionale Anforderungen

Neben den funktionalen Anforderungen sind weitere Punkte entscheidend für eine langfristige Praktikabilität. Die Nutzer der Plattform stehen im Mittelpunkt bei der Auswahl und Verwendung eines geeigneten Modells. Im Bezug zur **Benutzerfreundlichkeit (NFA_01)** müssen unterschiedliche Fähigkeiten der Nutzer bei der Informationspreisgabe und Anpassungsoptionen Berücksichtigung finden. Zudem spielt die **Performance (NFA_02)** eine große Rolle bei der Wahl eines passenden Modells. Auch wenn große und komplexe Datensätze zum Training genutzt werden, sollen Trainings- und Evaluationszeit bei hoher Qualität möglichst gering gehalten werden. Ebenso muss die vorhandene Rechenleistung angemessen genutzt werden. Weniger Rechenintensive Modelle erlauben eine höhere Anzahl an zu trainierenden und evaluierenden Modellen. Die **Zuverlässigkeit (NFA_03)** nimmt durch eine erhöhte Verfügbarkeit zu.

Des Weiteren soll die Möglichkeit bestehen weitere private Modelle der Plattform hinzuzufügen (**Erweiterbarkeit (NFA_04)**). Die Einführung von Modularität fördert hierbei zusammen mit einer ausführlichen **Dokumentation (NFA_05)** die Wiederverwendbarkeit. Neben einem kommentierten Programmcode dienen Beispiele und voreingestellte Parameter dazu, die Funktionsweise des Modells verständlicher zu gestalten. Darüber hinaus darf die **Sicherheit (NFA_06)** nicht vernachlässigt werden. Unbefugte Dritte dürfen nicht auf das Training des Modells zugreifen bzw. Änderungen vornehmen können.

4.2 Zielsetzung und Forschungsfrage

Abgeleitet aus den Anforderungen an einen Reproduktion-Service mit Fokus auf Privatsphäre ergeben sich Ziele und Forschungsfrage der Thesis.

Zielsetzung:
Aus einem schützenswerten tabellarischen Datensatz wird ein vollständig neuer Datensatz mit gleichen Eigenschaften generiert, der ohne Bedenken an Dritte weitergegeben werden kann.

Die übergeordnete Zielsetzung lässt sich insbesondere in zwei Hautziele unterteilen. Zum einen sollen qualitativ hochwertige Daten mit gleichen Eigenschaften generiert werden (**Z1**) und zum anderen die Privatsphäre der ursprünglichen Daten geschützt werden (**Z2**). Desgleichen sind die unterschiedlichen Vorerfahrungen der Nutzer zu berücksichtigen und entsprechend Informationsbedarf sowie Änderungsmöglichkeiten anzupassen (**Z3**). Der Trainingsdatensatz für die Experimente dieser Thesis muss so gewählt sein, dass er realitätsnahe Verteilungen und Eigenschaften widerspiegelt (**Z4**).

Die Umsetzung der Hauptziele Z1 und Z2 erfolgt durch die Verwendung eines GANs unter Einbezug von Differential Privacy. Aus diesem Grund fokussiert sich die zentrale Forschungsfrage auf das für den privaten Service bereitzustellende Modell.

Forschungsfrage:
Welches Generative Adversarial Network eignet sich für eine adäquate Synthese sensibler tabellarischer Daten unter Berücksichtigung von Differential Privacy?

Die zentrale Forschungsfrage wird mit Hilfe der Evaluationen von drei Teilaspekten beantwortet:

Frage 1 Performance: Wie viel Zeit benötigt das Modell für die Synthese von Daten?

Frage 2 Datenqualität: Inwiefern entsprechen die Eigenschaften der vom Modell generierten Daten denen der Trainingsdaten?

Frage 3 Privatsphärenschutz: Wie sicher sind die vom Modell generierten Daten gegenüber Angriffen?

4.3 Überblick der Experimente

Die definierten Teilaspekte werden anhand der in Kapitel 3.1 vorgestellten GANs (1) DPGAN, (2) PATE-GAN, (3) CTAB-GAN+ und (4) DP-CGANS untersucht. Die Experimente werden mit Hilfe von zwei Datensätzen aus dem Bereich Smart City durchgeführt (siehe Kapitel 4.4) und die generierten Daten mittels der in Kapitel 4.7 beschriebenen Evaluations-Metriken analysiert.

Zusätzlich zum Vergleich der Modelle und Datensätze wird bei der Evaluation zwischen vier Privatsphären Budgets (ϵ) unterschieden. Die Auswahl der Größe des ϵ-Wertes geht sowohl aus den Forschungsarbeiten der jeweiligen Modelle hervor als auch aus einer Zusammenstellung von Beispielen namhafter Tech-Unternehmen und US-Behörden. Das National Institute of Standards and Technology der USA (NIST) [62] fasst die aktuellen Erfahrungen wie folgt zusammen:

- ϵ-Werte im Bereich $\mathbf{0 < \epsilon \leq 5}$ garantieren einen starken Schutz der Privatsphäre, der als konservativ angesehen wird.
- Zunehmende Erfahrungen zeigen jedoch, dass auch ϵ-Werte im Bereich $\mathbf{5 < \epsilon \leq 20}$ in vielen Situationen einen ausreichend hohen Schutz an Privatsphäre bieten.
- Auch ϵ-Werte $>$ **20** können einen sinnvollen Schutz an Privatsphäre ermöglichen. Allerdings bedarf es weiterer Erfahrungen zur präzisen Einschätzung von höheren ϵ-Werten.

Um den Einfluss der verschiedenen Privatsphären Budgets zu verdeutlichen, werden die Daten mit den ϵ-Werten = 3, 10, 50 (mit Privatsphärenschutz) sowie $\epsilon = \infty$ (kein Privatsphärenschutz) generiert. Während die ϵ-Werte mit Privatsphärenschutz die Anzahl an Epochen begrenzen, ist bei der Generierung ohne Privatsphärenschutz die Anzahl an Epochen auf 500 bzw. 400 (CTAB-GAN+ und AGMA Datensatz) festgelegt.

Zusammenfassend ergeben sich folgende zu untersuchende **Kriterien**:

1. Modelle: DPGAN, PATE-GAN, CTAB-GAN+, DP-CGANS (DP-SGD vs. PATE)
2. Privatsphären Budget: 3, 10, 50, ∞ (DP vs. kein Privatsphärenschutz)
3. Datensätze: Energie, AGMA (simulierte vs. reale Daten)

Die Benennung der generierten Daten entspricht: *Modell_ Datensatz_ ϵ-Wert*

4.4 Smart City Datensätze

Das Forschungsprojekt DaFne konzentriert sich auf die Datensynthese im Bereich von Smart City. „Smart City“ impliziert den digitalen Wandel auf Infrastrukturen der Grundversorgung einer Region wie z.B. Beförderungsmöglichkeiten, Wasser- und Energieversorgung oder Müllbeseitigung. Neben der Digitalisierung der einzelnen Sektoren stehen auch deren Vernetzungen im Fokus. Der Einsatz Intelligenter Informations- und Kommunikationstechnologien (IKT) soll dabei zu einer ökonomisch, ökologisch und sozial nachhaltigen Region beitragen [14].

In Übereinstimmung mit dem Forschungsprojekt verwendet auch die Thesis Fallbeispiele aus dem Bereich Smart City. Hierbei handelt es sich um einen simulierten Datensatz mit Schwerpunkt Energieverbrauch einzelner Haushalte sowie um einen leicht modifizierten realen Datensatz zum Thema Mobilitätsströme und Tagesaktivitäten einzelner Personen. Die Nutzung eines simulierten sowie eines realen Datensatzes unterstützt den Vergleich von GANs bei unterschiedlicher Komplexität und Realitätsnähe, wobei die Zeilenanzahl beider Datensätze auf 40.000 begrenzt wird. Das im Zusammenhang mit der Generierung (simulierte Daten) stehende Jupyter Notebook ist der Thesis beigefügt (siehe Anhang A.1).

4.4.1 Energieverbrauch pro Haushalt: simulierter Trainingsdatensatz

Im Sektor smarte Energieversorgung existieren zahlreiche Use Cases bei denen Energieverbrauchsdaten benötigt werden. Sowohl Energieunternehmen als auch Verbraucher profitieren von intelligenten Technologien. Beispielsweise können Energie Engpässe und allgemeiner Bedarf vorhergesagt und somit die Zuverlässigkeit des Netzes verbessert werden. Auch eine effizientere Nutzung erneuerbarer Energien oder intelligenter Gebäudemanagementsysteme für eine erhöhte Überwachung des Verbrauchs einzelner Ressourcen ist vorstellbar.

Der simulierte Energieverbrauch pro Haushalt beschränkt sich auf simple Verteilungen einzelner Spalten und Zeilen. Pro Haushalt wird eine Zeile generiert, die Informationen zur befragten **Person** (z.B. Alter, Geschlecht, Berufsabschluss), **Haushaltsdaten** (z.B. Wohnsituation, Personenanzahl, Nettoeinkommen) und konkrete **Energieverbrauchsdaten** (z.B. Stromverbrauch, Raumwärme) beinhaltet. Insgesamt besteht der simulierte

Datensatz aus 17 Spalten, die sich aus acht diskreten (kategorialen) sowie neun kontinuierlichen (numerischen) Spalten zusammensetzen. Tabelle 4.2 gibt einen Überblick der Energieverbrauchsdaten.

Spaltenname	**Beschreibung**	**Datentyp**	**Quelle**
Person:			
Geschlecht	Weiblich | Männlich	Kategorisch	[72]
Alter	$25 \leq$ Alter ≤ 80	Numerisch	[73]
Familienstand	Ledig, Verheiratet usw.	Kategorisch	[74]
Bildungsabschluss	Ohne Schulabschluss, Realschulabschluss usw.	Kategorisch	[74]
Beruf	Selbständiger, Angestellter usw.	Kategorisch	[74]
Haushalt:			
Wohnsituation	Miete, Eigenes Haus usw.	Kategorisch	[44]
Personenanzahl	$1 \leq$ Anzahl ≤ 5 (zur Vereinfachung: 5+ ist gleich 5)	Numerisch	[74]
Nettoeinkommen	Unterteilt nach Werten z.B. <500, 2.00 - 2.500	Kategorisch	[74]
Gemeindegröße	Unterteilt nach Werten z.B. 20.00 - 50.000, >= 500.000	Kategorisch	[74]
Bundesland	Niedersachsen, Hamburg, Bayern usw.	Kategorisch	[74]
Verbrauchsdaten:			
Stromverbrauch	Ø Verbrauch je Person: 1PH = 1.978 | 2PH = 1.626 ab 3PH = 1.442 (Variabilität von 10%)	Numerisch	[71]
Energieverbrauch	Ø Verbrauch je Person: 1PH = 11.785 | 2PH = 9.340 ab 3PH = 6.915 (Variabilität von 10%)	Numerisch	[70]
Raumwärme	70,3% des Energieverbrauchs (Variabilität von 10%)	Numerisch	[70]
Warmwasser	14,7% des Energieverbrauchs (Variabilität von 10%)	Numerisch	[70]
Sonstige Prozesswärme	5,59% des Energieverbrauchs (Variabilität von 10%)	Numerisch	[70]
Sonstiger Betrieb Elektrogeräte	8% des Energieverbrauchs (Variabilität von 10%)	Numerisch	[70]
Beleuchtung	1,41% des Energieverbrauchs (Variabilität von 10%)	Numerisch	[70]

Tabelle 4.2: Beschaffenheit der Energieverbrauchsdaten

Während die Verteilungen der einzelnen Spalten auf den angegebenen Quellen basieren, wird die Abbildung von Korrelationen zwischen den Spalten in den meisten Fällen vernachlässigt. Lediglich die Spalten Strom- und Energieverbrauch werden in Abhängigkeit zur Anzahl an Personen im Haushalt berechnet. Die Verbrauchshöhe pro Person variiert entsprechend der zugrundeliegenden Statistik. Darüber hinaus wird der durchschnittliche Verbrauch mit einer Varianz von bis zu 10% generiert. Der Energieverbrauch untergliedert sich in Raumwärme, Warmwasser, sonstige Prozesswärme, sonstiger Betrieb von Elektro-

geräten sowie Beleuchtung. Auch die Zusammensetzung der Unterkategorien basiert auf Daten der Realität mit leichten Abweichungen zur durchschnittlichen Verteilung.

Auch wenn die einzelnen Spalten auf realen Verteilungen basieren, ist zu berücksichtigen, dass die Statistiken häufig nur stark aggregierte Informationen wie Durchschnittswerte preisgeben. Des Weiteren werden die Verbrauchsdaten ausschließlich auf Basis der Personenanzahl im Haushalt generiert. Weitere relevante Einflussfaktoren wie z.B. Wohnsituation, Beruf oder Nettoeinkommen werden vernachlässigt.

4.4.2 AGMA Daten: realer Trainingsdatensatz

Im Kontrast zu den vereinfachten Energieverbrauchsdaten, bestehen die AGMA Daten aus realen Befragungen zu Tagesaktivitäten und Mobilitätsströmen. Im Smart City Kontext können Daten dieser Thematik insbesondere bei einer smarten Stadtplanung und bei smarten Transportwegen nützlich sein. Use Cases im Bereich der smarten Stadtplanung betreffen beispielsweise die Standortwahl neuer Einkaufsfilialen, Bildungsstätten oder Gesundheitszentren. Ein anderes Beispiel betrifft eine verbesserte Beleuchtung und Müllentsorgung öffentlicher Plätze oder Straßen. Smarte Transportwege beziehen sich sowohl auf den privaten als auch auf den öffentlichen Verkehr. Verkehrsflusssteigerung, intelligente Parkplatzverwaltung oder eine optimierte Integration verschiedener Verkehrsträger entsprechen beispielhafter Anwendungsfelder.

Hinter der Abkürzung AGMA verbirgt sich die Arbeitsgemeinschaft Media-Analyse e.V. [3], ein Forschungsverbund bestehend aus mehr als 200 Unternehmen der Werbe- und Medienwirtschaft. Gemeinsam verfolgen diese das Ziel Leistungswerte für die Nutzung von Werbeträgern zu schaffen. Die dieser Thesis zur Verfügung stehenden Daten stammen aus ihrer „Media-Analyse: Out of Home“. Mit dem Ziel eine Grundlage zur Planung von Außenwerbung bereitzustellen, werden GPS-Messungen sowie Befragungen zur Demographie und Mobilität erhoben.

Im Fokus dieser Arbeit stehen die Daten aus der Befragung, da diese primär schützenswerte Daten einzelner Personen beinhalten. Nachdem redundante Spalten (z.B. Duplikate, Aggregation) entfernt wurden, verbleiben 67 relevante Spalten. Diese teilen sich in 57 diskrete (kategoriale) sowie zehn kontinuierliche (numerische) Spalten auf. Kleine Modifikationen seitens AGMA wie die Entfernung von fehlenden Werten sowie die Aggregation einzelner Spalten verzerren den Umgang mit realen Daten leicht, sind aber vorerst vernachlässigbar.

Anhand der Spalteninhalte ergeben sich acht übergeordnete Kategorien:

1. **Angaben zur Person**: z.B. Geschlecht, Alter, Bildung, Beruf
2. **Angaben zum Haushalt**: z.B. Ort, Nettoeinkommen, Personenanzahl
3. **Häufigkeit an Einkäufen**: z.B. in einem Supermarkt, Drogeriemarkt, Baumarkt
4. **Häufigkeit an Freizeitaktivitäten**: z.B. Nutzung von Medien, Tätigkeiten außer Haus, Reisen, Sport treiben, Raucher, Biertrinker
5. **Transportmittel**: z.B. Häufigkeit der Nutzung von Auto, Fahrrad, Bahn, Flugzeug
6. **Bewertungen zu unterschiedlichen Aussagen**: z.B. „Für besondere Qualität gebe ich gern mehr aus“, „Werbung ist eigentlich ganz hilfreich für den Verbraucher“
7. **Dauer außer Haus - Schätzung (Wegezeit)**: z.B. Montags, Samstags
8. **Daten zum Interview**: Monat und Jahr

Die exakt verwendeten Spalten werden im Anhang A.3 in der genannten Gruppierung aufgeführt. Um einen fairen Vergleich zu den simulierten Daten zu ermöglichen, werden die über 75.000 Teilnahmen auf zufällig ausgewählte 40.000 Zeilen begrenzt.

4.5 Überblick der verwendeten Modelle

Die bereits in Kapitel 3.1 eingeführten Modelle (1) DPGAN, (2) PATE-GAN, (3) CTAB-GAN+ sowie (4) DP-CGANS werden in den Experimenten verglichen und auf Grundlage dessen wird die Forschungsfrage beantwortet. Tabelle 3.1 fasst die grundlegenden Eigenschaften sowie Unterschiede zwischen den Modellen zusammen. Ergänzend um die allgemeine Vorstellung werden in diesem Kapitel tiefgehende Architekturkonzepte und relevante Parameter kompakt erklärt. Im Wesentlichen lassen sich die Parameter in die drei übergeordneten Kategorien Netzwerk Architektur, Netzwerk Training sowie Privatsphärenschutz einteilen. Eine detailreiche Übersicht aller wichtigen Parameterwerte befindet sich im Anhang A.4.

4.5.1 Netzwerk Architektur

In den Bereich Netzwerk Architektur fallen verwendete Verfahren zur Datenvorverarbeitung, Anzahl an Hidden Schichten und Knoten, Aktivierungsfunktionen sowie Regularisierungen der Netze.

Während DPGAN und PATE-GAN eine vereinfachte Vorverarbeitung der Daten - bestehend aus einer Skalierung (Bereich [0,1]) für kontinuierliche sowie One-Hot-Kodierung für diskrete Spalten - nutzen, profitieren CTAB-GAN+ und DP-CGANS von den Weiterentwicklungen im Bereich tabellarischer Datensynthese (siehe Kapitel 2.2.4). Beide Modelle wenden die Mode-Specific Normalization auf die kontinuierlichen sowie zusätzlich zur One-Hot-Kodierung das Training-by-Sampling auf die diskreten Daten an.

Um ein tieferes Verständnis möglicher Architekturvarianten zu erhalten, erfolgt eine beispielhafte Vorstellung der Architektur des DP-CGANS. Wie Abbildung 4.2 zeigt, besitzen Generator sowie Diskriminator zwei Hidden Schichten, die jeweils aus 256 Knoten bestehen. Der Generator verwendet in den Hidden Schichten ReLU als Aktivierungsfunktion mit Unterstützung einer Batch-Normalization. Die Batch-Normalization beschleunigt und stabilisiert das Netztraining, indem sie die Daten vor der Anwendung der Aktivierungsfunktion normalisiert und somit das Problem eines verschwindenden Gradienten minimiert. Die Output Schicht des Generators nutzt Tangens Hyperbolicus und Softmax als Aktivierungsfunktionen, um sowohl kontinuierliche als auch diskrete Daten erzeugen zu können. Die Hidden Schichten des Diskriminators beinhalten die LeakyReLU Aktivierungsfunktion sowie Dropout zur Regularisierung. Indem Dropout zufällig Verbindungen zwischen den Knoten ausschaltet, wird ein Overfitting vermieden. Das Netz wird gezwungen verschiedene Teilkombinationen seiner Knoten zu berücksichtigen und damit eine zu starke Anpassung an die Trainingsdaten verhindert. [77]

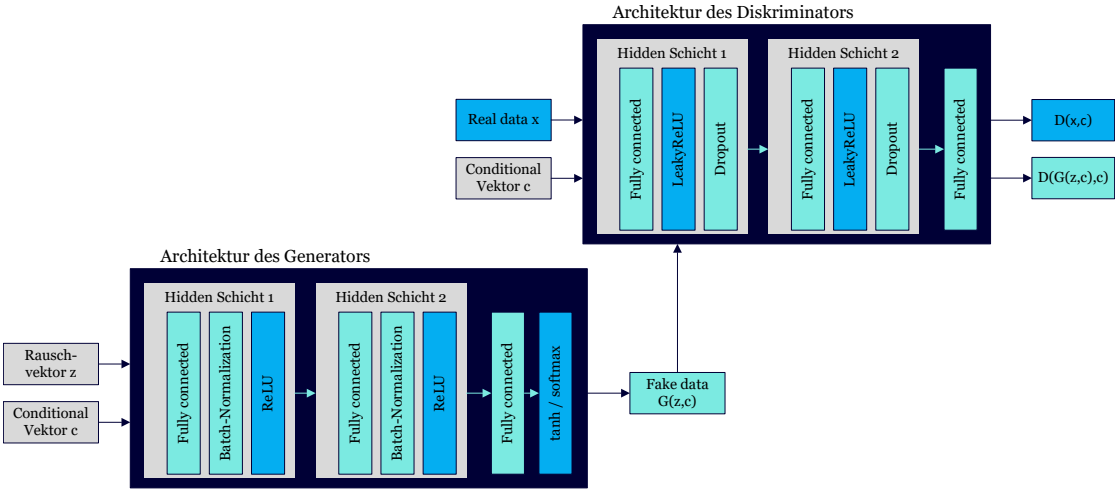

Abbildung 4.2: Architektur des DP-CGANS (in Anlehnung an [77])

CTAB-GAN+ integriert ebenfalls die Aktivierungsfunktionen ReLU und LeakyReLU. Die ergänzte Komponente C (Klassifikator oder Regressor) gleicht dem Aufbau des Diskriminators. Auch diese macht Gebrauch von der Aktivierungsfunktion LeakyReLu in Verbindung mit dem Dropout. Im Gegensatz zum DP-GANS besteht der Diskriminator sowie das zusätzliche Modell aber nicht aus zwei, sondern aus vier Hidden Schichten. DPGAN und PATE-GAN besitzen im Generator sowie Diskriminator jeweils eine Hidden Schicht, die beide die ReLU Aktivierungsfunktion nutzen. Dropout und Batch-Normalization finden bei beiden Modellen keine Anwendung. Die Anzahl an Lehrermodellen des PATE-GAN ist zudem auf zehn begrenzt.

4.5.2 Netzwerk Training

Das Netzwerk Training beinhaltet Informationen zur Epochenanzahl, Batchgröße, Optimierungsalgorithmus sowie Lernrate. Die Epochenanzahl liegt beim Training ohne Privatsphäre bei 500 (400 beim AGMA Training mit CTAB-GAN+), andernfalls ist die Anzahl vom erlaubten Privatsphären Budget abhängig. Die Batchgröße beträgt bei den Modellen DPGAN, CTAB-GAN+ sowie DP-CGANS in der Regel ebenfalls 500. Bei einer Anzahl von 40.000 Datenzeilen ergeben sich 800 Trainingsiterationen pro Epoche, die auf Grund einer hohen Trainingsdauer beim DP-CGANS bei einem Privatsphären Budget von 50 auf 400 (Batchgröße 1.000) reduziert werden. Darüber hinaus gleicht die Eingabe des Diskriminators beim DP-CGANS die eines PacGans [53]. Er trifft seine Entscheidung über real oder fake demnach nicht anhand eines Datensatzes, sondern erhält jeweils zehn Datensätze derselben Klasse als Grundlage für seine Bewertung.

Fast alle Modelle verwenden außerdem den Optimierungsalgorithmus Adam. Neben der Berechnung adaptiver Lernraten je Parameter beschleunigt Adam das Konvergieren des Netzes und gilt als weit verbreiteter Optimierungsalgorithmus. Die zugehörige Lernrate und der Weight Decay beeinflussen die Geschwindigkeit und Leistung des Modells.

4.5.3 Privatsphärenschutz

Zusätzlich zum Privatsphären Budget und der Fehlerwahrscheinlichkeit werden Noiseart, DP-Accountant, sowie die Größe von Sigma im Privatsphärenschutz aufgelistet. Sigma entspricht hierbei dem Gaussian Noise Variance Multiplier, der die Stärke des dem Gradienten hinzugefügten Rauschen bestimmt. Je größer der Wert, desto höher das Rauschen und so mehr Trainingsiterationen sind bei gleichem Privatsphären Budget möglich.

4.6 Modifikationen am Programmcode

Nachfolgend werden die verwendeten Programmcode-Repositorys der einzelnen Modelle genannt sowie vorgenommene Modifikationen aufgeführt. Die Programmcodes des **DP-GANs** sowie **PATE-GANs** stammen aus einer Zusammenführung verschiedener Generativer Modelle [13]. Im Unterschied zu ihren ursprünglichen Veröffentlichungen bauen sie auf einer bedingten Architektur auf. Auf Basis einer zu Beginn festgelegten Spalte werden alle anderen Spalteninhalte generiert. In den Experimenten wird auf den Spalten „**Wohnsituation**“ (Energie) und „**BIK-Regionstyp**“ (AGMA) trainiert. Darüber hinaus wird beim DPGAN das RDP-Accountant anstelle des Moment Accountant angewendet. Vom Autor zur Verfügung gestellte Beispieldaten liegen bereits in vorverarbeiteter Version im Repository. Explizite Klassen zur Vor- und Nachbearbeitung der Daten sind nicht vorhanden und werden entsprechend ergänzt. Die Daten der kontinuierlichen Spalten werden auf den Bereich null bis eins skaliert und die diskreten Daten mit Hilfe der One-Hot-Kodierung verarbeitet.

Die Datengenerierung mit den Modellen **CTAB-GAN+** [79] und **DP-CGANS** [76] erfolgt auf Grundlage der primären Repositorys. Der CTAB-GAN+ Programmcode wird um eine Main-Klasse erweitert. Diese unterstützt dabei Konfigurationen vereinfacht vorzunehmen und das Modell aus dem Code zu starten. Die Komponente C klassifiziert in den Experimenten die Spalten „**Wohnsituation**“ (Energie) und „**Nutzung des Verkehrsmittels U-Bahn, S-Bahn oder Regionalbahn in der Region**“ (AGMA). CTAB-GAN+ und DP-CGANS werden in ihrem ursprünglichen Programmcode für eine bestimmte Anzahl an Epochen trainiert. Da die Modelle in den Experimenten jedoch so lange trainiert werden sollen, bis sie ein definiertes Privatsphären Budget erzielen, wird die Kontroll-Variable Target-Epsilon eingeführt. Beim Erreichen dieser Ziel-Variable endet das Modelltraining und die Daten werden mit aktuellem Stand des Generators generiert. Des Weiteren werden automatisiert für alle vier Modelle CSV-Dateien angelegt, die die Trainingsdauer und verbrauchten Privatsphären Budget je Epoche dokumentieren.

4.7 Evaluations-Metriken

Mit dem primären Ziel qualitativ hochwertige Daten unter Berücksichtigung von Privatsphäre zu generieren, liegt der Fokus der Evaluation auf Datenqualität und Sicherheit der verwendeten sensitiven Trainingsdaten. Darüber hinaus spielt die Performance der Modelle eine entscheidende Rolle bei der Modellwahl für die DaFne Plattform.

4.7.1 Metriken zur Datenqualitätskontrolle

Die gewählten Metriken zur Prüfung der Datenqualität entstammen aus der frei verfügbaren Python-Library „Synthetic Data Metrics (SDMetrics)“, die Teil des Synthetic Data Vault Projektes ist [18]. SDMetrics unterstützt den Vergleich von realen und generierten Daten anhand unterschiedlicher Metriken, die teils visuell aufbereitet und in Reports gebündelt werden. Die einzelnen Metriken lassen sich zudem in die folgenden Arten untergliedern:

- **Single Column**: Prüfung einer Spalte (real vs. generiert)
- **Column Pairs**: Korrelationsvergleich zwischen zwei Spalten
- **Single Table**: Analyse einer gesamten Tabelle
- **Multi Table**: Untersuchung des Zusammengangs mehrerer Tabellen (hier irrelevant)
- **Sequential**: Kontrolle sequentieller Datenzeilen (hier irrelevant)

Neben der Art der Metrik gibt es Unterscheidungen in ihrer Anwendbarkeit. Während sich einige Metriken nur für diskrete oder kontinuierliche Datenspalten eignen, ermöglichen andere die Analyse beider Eigenschaften. In der Thesis werden der Quality sowie Diagnostic Report durchgeführt. Beide Reports enthalten sowohl Metriken für die Untersuchung diskreter als auch kontinuierlicher Datenspalten. Zur Ausführung werden die realen und generierten Daten sowie ihre Metadaten benötigt. Tabelle 4.3 gibt eine Übersicht zu den in den Reports inkludierten Metriken samt Beschreibung und Eigenschaften. Generell sind die Ergebnisse aller Metriken auf den Bereich 1.0 (beste Leistung) bis 0.0 (schlechteste Leistung) skaliert. Im Anschluss an die allgemeinen Reports werden Datenspalten mit auffällig schlechtem Ergebnis tiefgründiger analysiert.

Metrikname	Beschreibung	Metrikart	Datenart
Quality Report			
KS Complement	Kolmogorov-Smirnov-Statistik vergleicht die Wahrscheinlichkeitsverteilungen der numerischen Werte.	Single Column	Numerisch
TV Complement	Total Variation Distance berechnet den Unterschied der Häufigkeiten jeder Kategorie.	Single Column	Kategorisch
Correlation Similarity	Pearson-Korrelationskoeffizient misst die lineare Beziehung zwischen zwei Spalten.	Column Pairs	2x Numerisch
Contingency Similarity	Berechnung der Ähnlichkeit eines Paares kategorischer Spalten mit Vergleich des realen sowie synthetischen Datenpaars.	Column Pairs	2x Kategorisch
Discretize numerical & Contingency Similarity	Aufteilung der numerischen Werte in Kategorien und anschließender Vergleich mittels Contingency Similarity.	Column Pairs	1x Kategorisch & 1x Numerisch
Diagnostic Report			
New Row Synthesis	Metrik überprüft, ob in den generierten Daten Zeilen existieren, die identisch mit Zeilen aus den realen Daten sind.	Single Table	Kategorisch & Numerisch
Range Coverage	Metrik misst, ob eine generierte Spalte den gesamten Wertebereich seiner realen Spalte abdeckt.	Single Column	Numerisch
Category Coverage	Metrik prüft, ob generierte Spalte alle Kategorien der realen Spalte im richtigen Verhältnis abdeckt.	Single Column	Kategorisch
Boundary Adherence	Metrik berechnet den Anteil der generierten Werte einer Spalte, die innerhalb der Grenzen (Min-Max) der realen Daten liegen.	Single Column	Numerisch

Tabelle 4.3: Verwendete Qualitäts-Metriken unterteilt in Reports [18]

4.7.2 Verfahren zur Überprüfung von Privatsphäre

Aufbauend auf den Ergebnissen der Qualitätskontrolle werden die als qualitativ hochwertig eingestuften generierten Daten in einem zweiten Schritt auf ihren Privatsphärenschutz überprüft. Für die Sicherheitsanalyse wird das von Giomi et al. [31] entwickelte frei verfügbare Framework „Anonymeter“ verwendet. Mit dem Ziel in synthetisierten tabellarischen Datensätzen verbleibende Datenschutzrisiken aufzudecken und zu bewerten, integrieren die Autoren die Hauptindikatoren für Anonymisierung gemäß Datenschutzbestimmungen wie der Europäischen Datenschutzgrundverordnung (DSGVO). Explizit beinhaltet Anonymeter verschiedene Angriffsalgorithmen, um Risiken der Identifizierung, Verknüpfbarkeit und Inferenz zu ermitteln.

1. Risiko der **Identifizierung (Singling-out)**: berechnet die Wahrscheinlichkeit, dass ein Angreifer eine bestimmte Person im Datensatz isolieren kann. Anonymeter überprüft in diesem Fall den Datensatz auf einzigartige Kombinationen von Ausprägungen, die auf eine Person zutreffen.
2. Risiko der **Verknüpfbarkeit (Linkability)**: berechnet die Wahrscheinlichkeit, dass ein Angreifer zwei oder mehrere Einträge aus verschiedenen Datensätzen, die zur selben Person gehören, verknüpfen kann. Anonymeter erhält zwei disjunkte Mengen an Ausprägungen und bewertet mithilfe des synthetischen Datensatzes, ob die Teildatensätze zu derselben Person gehören oder nicht.
3. Risiko der **Inferenz (Inference)**: berechnet die Wahrscheinlichkeit, dass ein Angreifer sensible Informationen über eine Person im Datensatz ableiten kann. Anonymeter analysiert diesbezüglich Korrelationen zwischen verschiedenen Spalten im Datensatz.

5 Evaluationsergebnisse

Bezugnehmend auf die drei Teilfragen zur Beantwortung der Forschungsfrage (definiert in 4.2) erfolgt die Präsentation der Evaluationsergebnisse. Nach einer Analyse der Modellperformance in Bezug auf Geschwindigkeit und Eigenschaften der Epochen, wird die Datenqualität auf unterschiedliche Merkmale geprüft und anschließend der Privatsphärenschutz mittels ML-Angriffen getestet. Zu Beginn jedes Unterkapitels werden aus den gewonnen Erkenntnissen der Kapitel 2 bis 4 Hypothesen (HT) aufgestellt. Ihre Definition unterstützt das Aufdecken von Schwachstellen sowie Stärken der einzelnen Modelle und offenbart unerwartete Ergebnisse. Abschließend werden die Hypothesen abgeglichen und die wichtigsten Resultate zusammengefasst. Alle im folgenden Kapitel gemachten Aussagen beziehen sich ausschließlich auf die durchgeführten Experimente und sind auf den Rahmen dieser Untersuchungen beschränkt. Um allgemeingültige Nachweise zu erbringen, bedarf es weiterer Experimente.

5.1 Modellperformance

Um die erste Teilfrage zu untersuchen, wird die Modellperformance bezüglich absoluter Trainingszeit und Dauer, Anzahl sowie Anstieg des Privatsphären Budgets pro Epoche verglichen. Insbesondere die Trainingsdauer besitzt einen großen Einfluss auf die Verwendung des ML-Modells.

Id	**Beschreibung**
HT_1.1	Je größer das Privatsphären Budget, desto zeitintensiver die Generierung.
HT_1.2	Je größer die Spaltenanzahl, desto zeitintensiver die Generierung.
HT_1.3	Die Dauer einer Epoche ist beim Modelltraining mit DP im Vergleich zum Modelltraining ohne DP erhöht.
HT_1.4	Die Anzahl an Epochen steigt mit dem Privatsphären Budget.

Tabelle 5.1: Hypothesen zur Modellperformance

Trainingsdauer Grundsätzlich demonstrieren alle vier Modelle, dass ein höheres Privatsphären Budget zu einer höheren Trainingszeit führt (**HT_1.1**). Bei gleichem Privatsphären Budget erweist sich das CTAB-GAN+ fast immer als schnellstes Modell. Lediglich bei der Generierung der synthetischen Energiedaten ohne Privatsphäre liegt das DP-CGANS vor dem CTAB-GAN+. Als besonders langsam trainierendes Modell offenbart sich das PATE-GAN. Allein für die Generierung der Energiedaten mit einem Privatsphären Budget von 50 benötigt das Modell über zehn Tage. Unter anderem aus diesem Grund wird auf die Generierung der AGMA Daten mit einem Privatsphären Budget von 50 beim PATE-GAN verzichtet.

Die Generierung der Energiedaten benötigt im Vergleich zu den AGMA Daten weniger Zeit und bestätigt somit im Rahmen der hier durchgeführten Experimente **HT_1.2**. Die Spannbreite der unterschiedlichen Dauer je Datensatz ist stark vom verwendeten Modell abhängig. Während beim CTAB-GAN+ nur wenige Minuten zwischen der Generierung der AGMA und Energie Daten liegen, zeigt das DP-CGANS Geschwindigkeitsprobleme bei der Synthese von Daten hoher Dimension. Folglich wird der AGMA Datensatz für das Training mit dem DP-CGANS auf 25 Spalten gekürzt.

Zeit pro Epoche Auch wenn die absolute Trainingsdauer ohne Differential Privacy nicht die kürzeste ist, deutet die benötigte Zeit pro Epoche auf ein deutlich schnelleres Training hin (**HT_1.3**). Des Weiteren fällt auf, dass alle Modelle bis auf das DP-CGANS bei unterschiedlich hohem Privatsphären Budget eine konstante Trainingsdauer je Datensatz pro Epoche besitzen. Die Trainingsdauer pro Epoche wird beim DP-CGANS durch das gewählten Privatsphären Budget beeinflusst. Mit Zunahme des Privatsphären Budgets steigt die benötigte Zeit pro Epoche stark an.

Anstieg des Privatsphären Budgets Im Gegensatz zur häufig konstanten Trainingszeit pro Epoche weisen alle vier Modelle eine Anstiegsreduktion des Privatsphären Budgets pro Epoche im Trainingsverlauf vor. Vor allem beim DP-CGANS führt dies zu einer verstärkten Verlängerung der Trainingsdauer. Hier treffen die im Verlauf steigende Zeit sowie der immer kleiner werdende Anstieg des Privatsphären Budgets pro Epoche aufeinander (vgl. Abbildung 5.1).

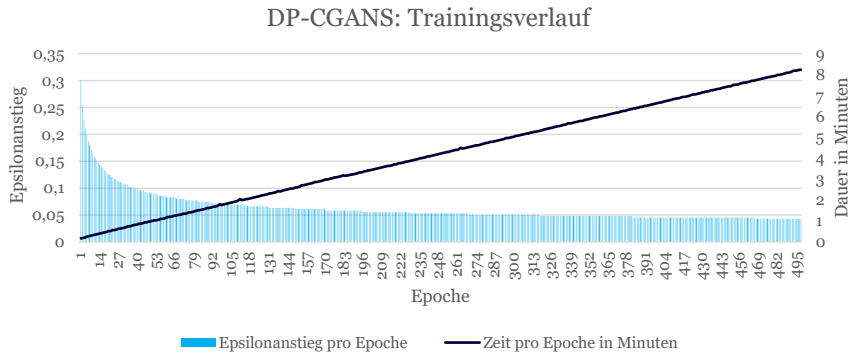

Abbildung 5.1: Dauer und ϵ-Anstieg je Trainingsepoche des DP-CGANS

Anzahl an Epochen Der immer kleiner werdende Anstieg des Privatsphären Budgets spiegelt sich in der Anzahl an Epochen wider. Mit größer werdendem Privatsphären Budget wächst die Anzahl gravierend **(HT_1.4)**. Auffällig ist ein großer Unterschied zwischen den Modellen, der durch die ebenfalls sich stark unterscheidenden Größen im Anstieg des Privatsphären Budgets pro Epoche entsteht (siehe Anhang A.5).

5.2 Datenqualität

Um sich der Teilfrage zur Datenqualität zu nähern, werden die Eigenschaften der generierten Daten mit den Eigenschaften der realen Daten unter Inanspruchnahme der verschiedenen Metriken (vorgestellt in 4.7.1) verglichen. Für eine verbesserte Übersicht sind die Ergebnisse des Quality Reports und Diagnostic Reports separat aufgeführt.

Id	Beschreibung
HT_2.1	Modelltraining ohne DP führt im Vergleich zum Modelltraining mit DP zu einer verbesserten Datenqualität.
HT_2.2	Je größer das Privatsphären Budget, desto besser die Datenqualität.
HT_2.3	Je komplexer der Datensatz, desto schlechter die Datenqualität.
HT_2.4	Kategoriale Spalten können im Vergleich zu kontinuierlichen Spalten besser abgebildet werden.
HT_2.5	Je geringer die Anzahl an unterschiedlichen Kategorien einer Spalte, desto besser die Datenqualität.

HT_2.6	Je schwächer die Korrelation zwischen zwei Spalten, desto besser die Datenqualität.

Tabelle 5.2: Hypothesen zur Datenqualität

5.2.1 Quality Report

Nach der Untersuchung eines zusammengefassten Wertes zur Gesamtqualität der Daten, erfolgt die Evaluation der einzelnen Bestandteile des Quality Reports. Die expliziten Metriken werden den Kategorien Beschaffenheit der einzelnen Spalten sowie Korrelationen zwischen Spalten zugeordnet und nachfolgend analysiert.

Gesamtqualität Aggregiert über alle Metriken des Quality Reports gibt die Gesamtqualität erste Hinweise auf verwendbare Daten. Aus der Abbildung 5.2 werden große Qualitätsunterschiede zwischen den DP-Ursprungsmodellen PATE-GAN und DP-GAN sowie den fortgeschrittenen Modellen CTAB-GAN+ und DP-CGANS unmittelbar sichtbar. Während das PATE-GAN nur ein Qualitätsniveau von 0,3 erreicht, erzielt das CTAB-GAN+ immer Ergebnisse über 0,8. Die definierten Hypothesen treten ausschließlich beim CTAB-GAN+ und DP-CGANS ein. Die Datenqualität beim DPGAN und PATE-GAN nimmt weder mit steigendem Privatsphären Budget zu (**HT_2.2**) noch weist der weniger komplexe Energie Datensatz eine höhere Qualität auf (**HT_2.3**).

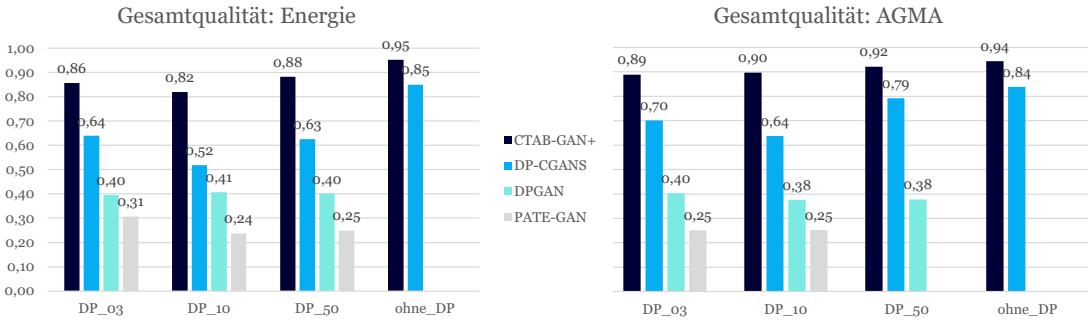

Abbildung 5.2: Gesamtqualität aller generierten Daten unterteilt nach Datensatz

Im Gegensatz dazu ist die Datenqualität beim CTAB-GAN+ und DP-CGANS vielversprechend. Insbesondere das CTAB-GAN+ überzeugt in allen Generierungen unabhängig vom Datensatz und Privatsphären Budget. Beide fortgeschrittenen Modelle erzielen die besten Ergebnisse beim Training ohne DP (**HT_2.1**) und erreichen zumeist auch beim Training mit DP eine hochwertige Datenqualität.

Wie vermutet ist die Qualität beim Training mit einem Privatsphären Budget von 50 am besten. Dennoch trifft die Hypothese **HT_2.2** nicht in allen Fällen zu. Bis auf das Training von CTAB-GAN+ mit den AGMA Daten, zeigt sich, dass die Qualität beim Training mit einem Privatsphären Budget von 10 schlechter abschneidet als die bei 3. Entgegen der Erwartung lassen sich zudem die AGMA Daten im Vergleich zu den weniger komplexen Energiedaten mit einer besseren Datenqualität generieren (**HT_2.3**).

Beschaffenheit der Spalten Die Beschaffenheit der einzelnen Spalten übersteigt die Gesamtqualität (vgl. Abbildung 5.3). Daraus lässt sich schließen, dass sich die Eigenschaften individueller Spalten einfacher erlernen lassen als ihre Korrelationen. Ansonsten gleichen die Befunde denen, die sich auf die Gesamtqualität beziehen. CTAB-GAN+ und DP-CGANS generieren eine deutlich bessere Qualität als PATE-GAN und DPGAN. Eine Vergrößerung des Privatsphären Budgets führt nicht immer zu einer erwarteten Qualitätsverbesserung (**HT_2.2**) und die AGMA Daten lassen sich in einer höheren Qualität abbilden als Energie Daten (**HT_2.3**).

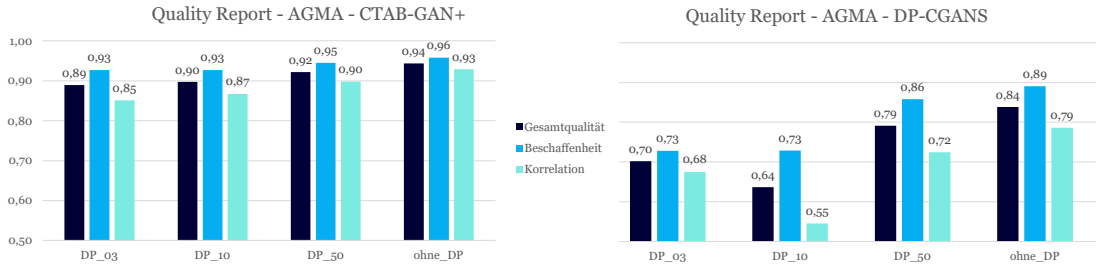

Abbildung 5.3: Ergebnisse des Quality Reports zu den Modellen CTAB-GAN+ und DP-CGANS

Ein detaillierter Blick in die explizit verwendeten Metriken zur Berechnung der Beschaffenheit der Spalten (vgl. Abbildung 5.4) enthüllt bei allen Datensätzen und Modellen eine große Differenz zwischen den Ergebnissen des TV Complements (kategorische Daten) und KS Complements (numerische Daten). Die Kategorien lassen

sich besser in der realen Wahrscheinlichkeitsverteilung reproduzieren als der Wertebereich bei den numerischen Spalten (**HT_2.4**). Angesichts der Struktur der Trainingsdaten wird ersichtlich, weshalb die AGMA Daten im Vergleich zu den Energiedaten insgesamt präziser repräsentiert werden können. Ein Verhältnis von 57:10 (Kategorisch: Numerisch) der AGMA Spalten steht dem Verhältnis von 8:9 der Energie Daten gegenüber (**HT_2.3**).

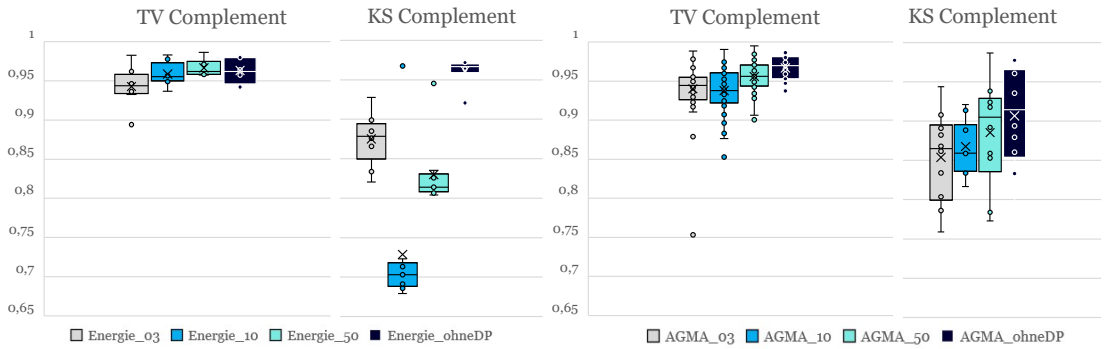

Abbildung 5.4: Vergleich der Metriken TV- und KS-Complement beim CTAB-GAN+

Sowohl bei den generierten Energiedaten als auch bei den AGMA Daten wird außerdem sichtbar, dass es den Modellen schwer fällt kategorische Daten mit hoher Ausprägungsanzahl zu generieren (**HT_2.5**). Die Modelle zeigen Schwierigkeiten die Spalte „Bundesland“, die die größte Anzahl an Ausprägungen in beiden Datensätzen besitzt, exakt abzubilden. Im Gegensatz dazu kann die Spalte „Geschlecht“ besonders zuverlässig vorhergesagt werden.

Korrelationen zwischen Spalten Wie im vorherigen Absatz angedeutet, erzeugen die Modelle bei der Abbildung von Korrelationen zwischen Spalten schlechtere Ergebnisse als bei der Reproduktion von unabhängigen Spalteneigenschaften. Mit Zunahme der Gesamtqualität zeichnet sich eine Reduktion dieser Differenz ab. Andernfalls veranschaulichen die Korrelationswerte bezüglich der verschiedenen Kriterien ein ähnliches Verhalten wie die zuvor untersuchten Metriken (vgl. Abbildung 5.3).

Auch in diesem Fall belegen die zugrundeliegenden Metriken eine Diskrepanz zwischen der Contigency Similaritiy und der Correlation Similarity. Im Kontrast zur Datenbeschaffenheit fällt auf, dass sich die Korrelationen zwischen den numerischen

Datenspalten (Correlation Similarity) am besten erlernen lassen. Ferner verdeutlichen die Ergebnisse, dass schwache Korrelationen zwischen den Spalten besser reproduziert werden können als hohe Korrelationen (**HT_2.6**).

Abbildung 5.5 zeigt die Korrelationen zwischen den numerischen Werten. Es wird deutlich, dass mit zunehmendem Privatsphären Budget die realen Korrelationen gezielter reproduziert werden. Darüber hinaus werden unterschiedlich ausgeprägte Korrelationen zwischen Energie (links) und AGMA (rechts) Daten sichtbar.

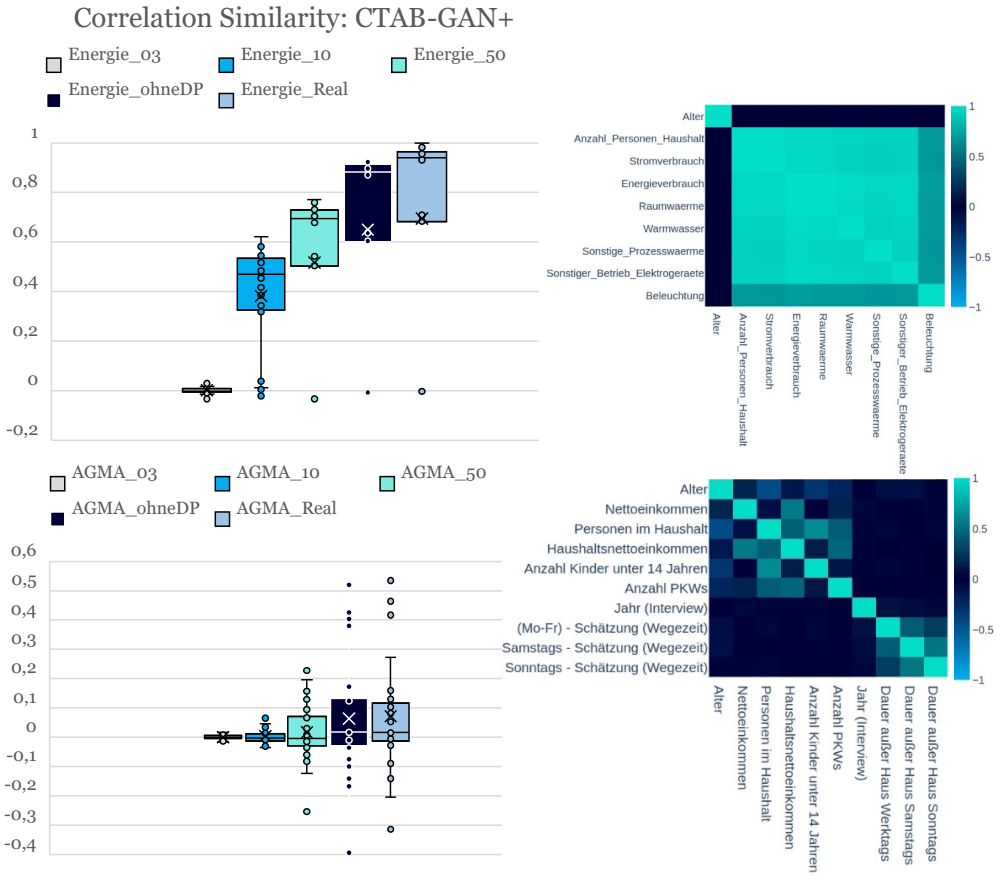

Abbildung 5.5: Korrelationen zwischen numerischen Daten des CTAB-GAN+

Bei der Contigency Similaritiy bedarf es einer Unterteilung in der Bewertung von Korrelationen zweier kategorischer Spalten und von Korrelationen zwischen einer kategorischen und numerischen Spalte. Vor allem treten hier Schwierigkeiten beim Abbilden der Korrelationen zwischen kategorischer und numerischer Spalten auf.

5.2.2 Diagnostic Report

Die Evaluation des Diagnostic Reports gibt Aufschluss über die Syntheseleistung der Modelle, überprüft inwiefern die Wertebereiche einzelner Spalten abgedeckt werden und ob die numerischen Daten ihre ursprünglichen Grenzwerte überschreiten.

Syntheseleistung Eine Eigenschaft, die alle vier Modelle erfüllen, ist die Fähigkeit zu 100% neue Daten zu generieren. Im Abgleich zwischen realen und generierten Daten lassen sich keine identischen Datensätze wiederfinden. Es existieren demnach in keinem Fall Kopien, die unmittelbare Rückschlüsse auf ein komplettes Datum aus den realen Daten zulassen könnten.

Abdeckung des Wertebereichs Auch bei der Abdeckung des Wertebereichs einzelner Spalten zeigen die meisten Modelle wenige Probleme. Insbesondere die Modelle DPGAN und CTAB-GAN+ decken den Wertebereich bei jeder Generierung zu über 90% ab. PATE-GAN umfasst den Wertebereich der AGMA Spalten ebenfalls zufriedenstellend, hat aber Herausforderungen den Wertebereich der Energie Spalten geeignet abzubilden. Beim DP-CGANS fallen die generierten Daten mit einem Privatsphären Budget von zehn auf. Im Verhältnis zu den übrigen Ergebnissen werden die Wertebereiche hier weniger gut abgedeckt (vgl. Abbildung 5.6).

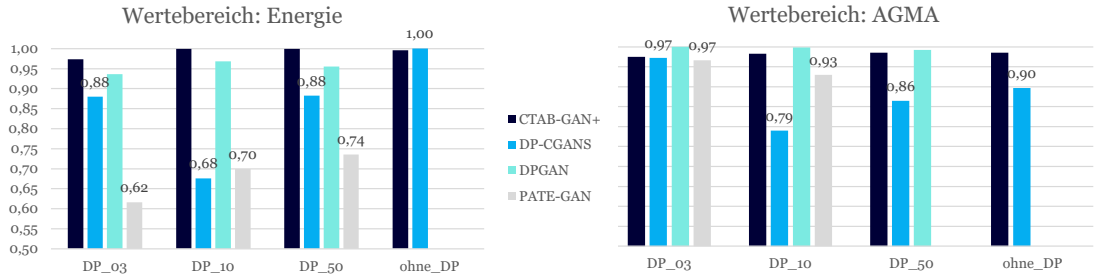

Abbildung 5.6: Abdeckung des Wertebereichs unterteilt nach Datensatz

Beim Abgleich der konkreten Metriken wird sichtbar, dass das PATE-GAN bei der Synthese der Energiedaten größere Schwierigkeiten mit dem Category Coverage als mit dem Range Coverage besitzt. Im Kontrast dazu weist das DP-CGANS beim Category Coverage nahezu immer eine Abdeckung von 100% auf, offenbart aber bei einem Privatsphären Budget von zehn Schwächen bei der Erfassung aller Werte des Range Coverages (**HT_2.4**).

Einhaltung von Grenzwerten Die größte Unterscheidung zwischen den ursprünglichen und fortgeschrittenen DP-Modellen liegt im Einhalten von Grenzwerten der numerischen Spalten (vgl. Abbildung 5.7). Während CTAB-GAN+ und DP-CGANS keine numerischen Daten außerhalb der Grenzbereiche der realen Datenbasis generieren, scheinen DPGAN und PATE-GAN die ursprünglichen Wertegrenzen vollständig zu ignorieren. Beispielsweise enthält die Spalte „Alter“ negative Werte und Maxima von über mehreren 1.000. Als Folge sind viele numerische Werte beim DPGAN sowie PATE-GAN unsinnig und ihre Spalten nicht verwendbar.

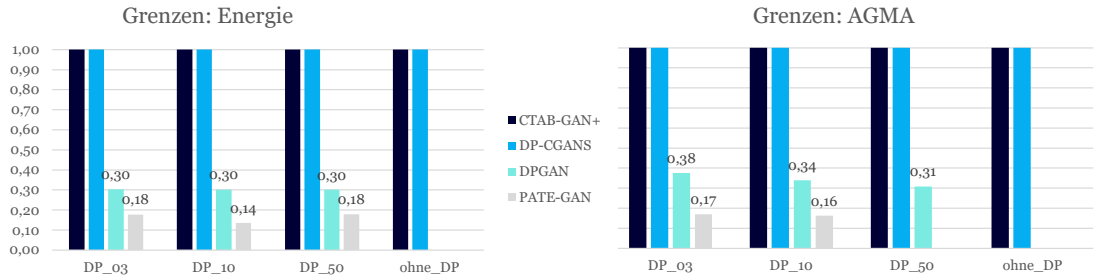

Abbildung 5.7: Einhaltung von Grenzwerten unterteilt nach Datensatz

5.3 Privatsphärenschutz

Die Analyse zum Privatsphärenschutz erfolgt mittels des in Kapitel 4.7.2 eingeführten Frameworks Anonymeter. Die integrierten Angriffs-Modelle werden genutzt, um die zwölf qualitativ hochwertigsten Datensätze bezüglich ihres Risikos gegenüber Identifizierung, Verknüpfbarkeit und Inferenz zu überprüfen. Explizit handelt es sich bei den ausgewählten Datensätzen um alle vom CTAB-GAN+ generierten Datensätze sowie die vom DP-CGANS generierten Datensätze mit einem Privatsphären Budget von 50 und ∞. Die Ergebnisse der Angriffs-Modelle enthalten jeweils drei Werte: das vorhergesagte Risiko (Mittelwert aller Angriffe) sowie die Minima und Maxima des 95%-Konfidenzintervalls. Um möglichst zuverlässige Einschätzungen zum Privatsphärenschutz zu geben, werden bei der Auswertung insbesondere die Maxima analysiert. Die konservative Haltung führt dazu, dass mit einer 95% Sicherheit die Risiken maximal so hoch sein werden, wie die berechneten Werte. Alle Experimente werden drei Mal durchgeführt und ihre Resultate anschließend gemittelt.

Id	Beschreibung
HT_3.1	Modelltraining mit DP führt im Vergleich zum Modelltraining ohne DP zu einem erhöhten Datenschutz.
HT_3.2	Je niedriger das Privatsphären Budget, desto höher der Datenschutz.
HT_3.3	Je mehr Informationen einem Angreifer zur Verfügung stehen, desto höher das Risiko, dass unbekannte Spaltenausprägungen ermittelt werden können.

Tabelle 5.3: Hypothesen zum Privatsphärenschutz

Abbildung 5.8 zeigt einen aggregierten Überblick zu den maximalen Risiken der einzelnen Angriffsformen. Während die oberen beiden Diagramme Ergebnisse zu den AGMA Daten präsentieren, berücksichtigen die beiden unteren die Energie Daten. Zu beachten sind die unterschiedlichen Wertebereiche der Y-Achse, die auf zum Teil unsichere Energie Daten hinweisen. Darüber hinaus wird deutlich, dass die Risiken zur Verknüpfbarkeit besonders niedrig sind und fast immer bei unter 1% liegen. Im Folgenden werden weitere Auffälligkeiten herausgestellt und analysiert.

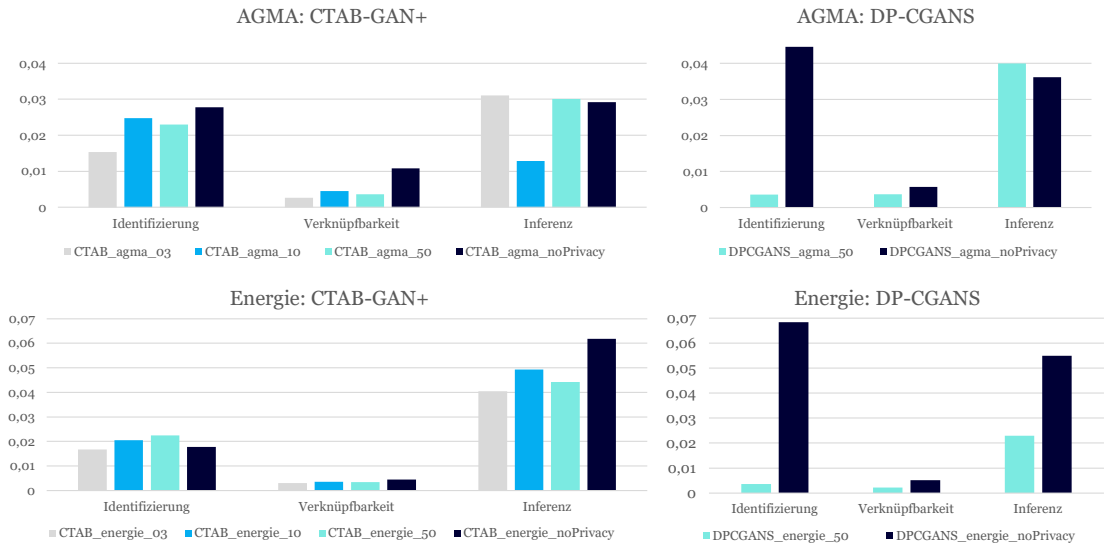

Abbildung 5.8: Gegenüberstellung von Risiken im Privatsphärenschutz nach Angriffsart

Identifizierung Basierend auf der Annahme, dass seltene Ausprägungen von Spalten in den realen Datensätzen ebenfalls selten in den generierten Daten vorkommen, wird mit dieser Angriffsart die Identifizierbarkeit einzelner Personen untersucht. Unterschieden wird zwischen der Uni- und Multi-Variante.

Mit dem Ziel Spalten zu entdecken, die viele verschiedene Ausprägungen (bspw. IDs, Adressen, Telefonnummer) besitzen, berechnet die Uni-Variante die Risiken anhand einer Spalte. Mit einer Anzahl von 750 Attacken je Durchlauf liegen die Risiken unabhängig vom Datensatz und Privatsphären Budget beim CTAB-GAN+ zwischen 1% und 3%. Hingegen weisen die Auswertungen zu den vom DP-CGANS generierten Daten hohe Differenzen zwischen den Daten auf, die mit und ohne Differential Privacy trainiert wurden. Die Ergebnisse befinden sich beim Training ohne DP bei über 4% (AGMA) bzw. 6% (Energie) und fallen beim Training mit einem Privatsphären Budget von 50 unter 1% (vgl. Abbildung 5.8).

Deutlich höher sind die Risiken für eine Identifikation bei der Analyse zusammenhängender Spalten. Für den AGMA und Energie Datensatz werden bei der Multi-Variante jeweils fünf, zehn und fünfzehn Spalten ausgewählt und ihre jeweiligen Ausprägungen verknüpft. Die verwendeten Spalten können im zugehörigen Jupyter Notebook (siehe Anhang A.1) eingesehen werden. Aufsteigend von fünf auf fünfzehn verknüpften Spalten zeigt sich, dass die Risiken einer Identifikation abnehmen. Maximale Wahrscheinlichkeiten von bis zu 23% sind beim Datensatz CTAB_agma_noPrivacy mit der Verknüpfung von fünf Spalten möglich. Bei gleichen Trainingsdaten und Modell, aber einem Privatsphären Budget von zehn, kann das Risiko um 10% gesenkt werden.

Abbildung 5.9 veranschaulicht die Risiken der Multi-Variante. Bei allen Variationen wird sichtbar, dass das Training ohne DP höhere Risiken im Vergleich zum Training mit DP birgt (**HT_3.1**). Jedoch kann nicht bewiesen werden, dass bei Abnahme des Privatsphären Budgets auch immer die Risiken minimiert werden (**HT_3.2**).

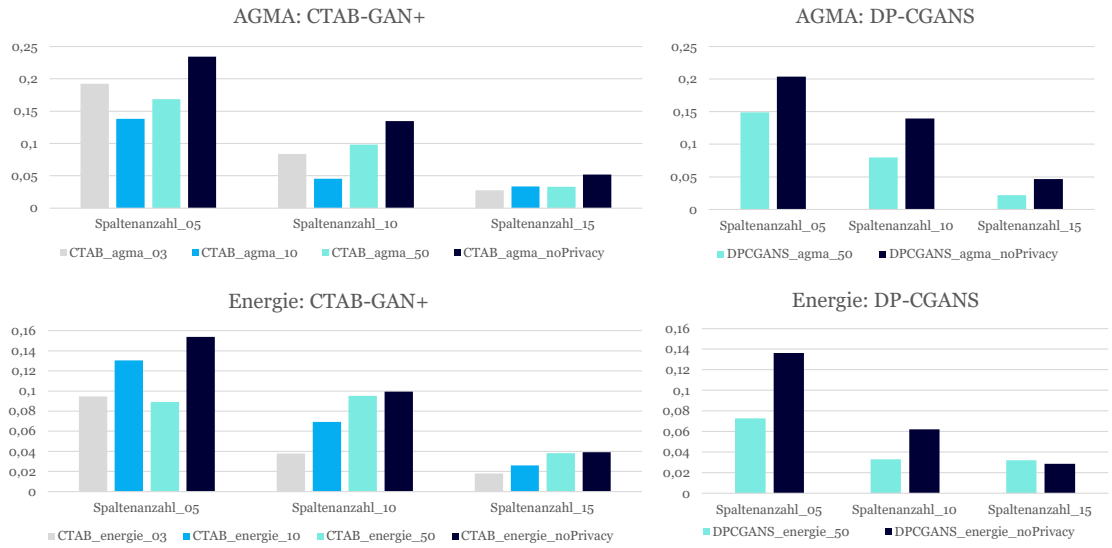

Abbildung 5.9: Risiken der Identifizierung unter Nutzung der Multi-Variante

Verknüpfbarkeit Bei Angriffen im Rahmen der Verknüpfbarkeit wird davon ausgegangen, dass dem Angreifer zwei Teile aus dem realen Datensatz vorliegen und dieser zusammen mit den generierten Daten versucht die Teile zu verknüpfen. Konkret sucht der Angreifer mittels des Nearest-Neighbor Algorithmus für jedes Datum im ersten Teil eine definierte Anzahl k an synthetischen Daten, die den Eigenschaften des jeweiligen Datums am ähnlichsten sind. Diese werden dann verwendet, um im zweiten Datenteil Datensätze zu finden, die derselben Person gehören könnten.

Für die Evaluation stehen dem Angreifer jeweils Teildatensätze bestehend aus drei, fünf oder acht Spalten zur Verfügung. Darüber hinaus wird die Anzahl an zu suchenden synthetischen Datensätzen ähnlicher Eigenschaften auf zwei, vier, sechs, acht sowie zehn festgelegt. Grundsätzlich fällt bei allen Experimenten auf, dass die Risiken begrenzt sind. Die Wahrscheinlichkeit Datensätze zu verknüpfen, liegt nahezu immer bei unter 1%. Auch spiegeln sich die Erkenntnisse zur Identifizierbarkeit in diesen Angriffen wider. Erneut sind die mit DP generierten Daten mit einzelnen Ausnahmen besser geschützt als die ohne (**HT_3.1**) und ein niedrigeres Privatsphären Budget weist entgegnen der Erwartungen nicht in jedem Fall auf besser geschützte Daten hin (**HT_3.2**).

Des Weiteren zeichnet sich ab, dass das Risiko mit der Anzahl vom Angreifer verwendeten synthetischen Datensätzen ansteigt (vgl. Abbildung 5.10). Tendenzen deuten zudem daraufhin, dass das Risiko der Verknüpfbarkeit zunimmt, wenn die realen Teilmengen eine größere Anzahl an Spalten beinhalten (**HT_3.3**).

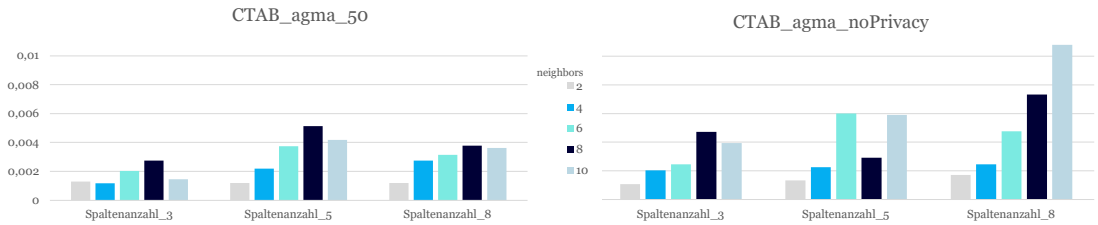

Abbildung 5.10: Risiken der Verknüpfbarkeit unterschiedlich großer Datensätze

Inferenz Bei den Angriffen zur Berechnung des Inferenz-Risikos wird davon ausgegangen, dass dem Angreifer Informationen zur Ausprägung einzelner Spalten vorliegen. Unter dieser Prämisse verfolgt der Angreifer das Ziel, weitere sensible Informationen zu einzelnen Personen herauszufinden. Ähnlich wie beim Risiko Verknüpfbarkeit wird der Nearest-Neighbor Algorithmus verwendet, um den synthetischen Datensatz nach Einträgen mit möglichst ähnlichen Ausprägungen durchzusuchen.

Auf Basis der Annahme, dass dem Angreifer alle Ausprägungen bis auf eine Spalte zur Verfügung stehen, wird für jede Spalte einzeln das Risiko der Inferenz berechnet. Je Datensatz werden anschließend die zehn Spalten mit dem größten Risiko gegenübergestellt. Große Unterschiede zeigen sich zwischen den generierten AGMA und Energie Daten. Während risikoreiche Spalten im AGMA Datensatz mit einer Wahrscheinlichkeit von über 10% vorhergesagt werden können, erreichen Spalten im Energie Datensatz nur ein Risiko von 4% (**HT_3.3**). Ein verhältnismäßig hohes Risiko bergen im AGMA Datensatz die Spalten: „Einkauf in Großmärkten“ sowie „Berufliche Flugzeugnutzung“. Diese Spalten teilen die Eigenschaft eine Ausprägung zu besitzen, die den Großteil der Vorkommen ausmacht. 91% der befragten Personen nutzen beruflich kein Flugzeug und 87% gehen selten bzw. nie in Großmärkten einkaufen (vgl. Abbildung 5.11).

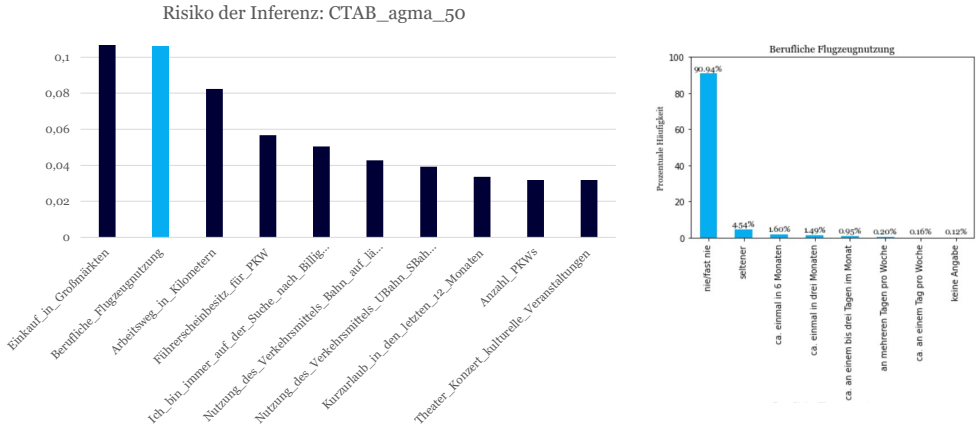

Abbildung 5.11: Bezüglich Inferenz gefährdete Spalten mit beispielhafter Eigenschaft der Spalte „Berufliche Flugzeugnutzung“

Werden die Ergebnisse der einzelnen Spalten zusammen betrachtet, zeigt sich, dass die Risiken aggregiert beim Energie Datensatz höher ausfallen. Wie in Abbildung 5.8 zu sehen, liegen sie bis auf eine Ausnahme über 4%, während sich die Risiken beim AGMA Datensatz des Öfteren bei und unter 3% befinden.

5.4 Zusammenfassung der Ergebnisse

Tabelle 5.4 fasst die Ergebnisse der untersuchten Hypothesen zusammen. Obgleich sich die meisten Hypothesen als gültig herausstellen, werden **HT_2.3** (Komplexität beeinträchtigt Qualität) widerlegt und **HT_2.2 & HT_3.2** (Privatsphären Budget beeinflusst Datenqualität und Datenschutz) nur teilweise bestätigt. Da vor allem letztere jedoch entscheidende Annahmen dieser These sind, werden ihre Ergebnisse im Kapitel 6 infrage gestellt. Ein Grund für die Entkräftung der Hypothese **HT_2.3** konnte bereits in den unterschiedlichen Verhältnisse von kontinuierlichen sowie diskreten Spalten in den Datensätzen identifiziert werden. Die Bestätigung von **HT_2.6** (Korrelationsstärke beeinflusst Datenqualität) unterstützt die Argumentation, denn auch in diesem Fall sind die Eigenschaften der kontinuierlichen Daten des Energie Datensatzes im Vergleich zum AGMA Datensatz schwieriger zu reproduzieren.

Erfüllt	Id	Beschreibung
		Modellperformance
✓	**HT_1.1**	Je größer das Privatsphären Budget, desto zeitintensiver die Generierung.
✓	**HT_1.2**	Je größer die Spaltenanzahl, desto zeitintensiver die Generierung.
✓	**HT_1.3**	Die Dauer einer Epoche ist beim Modelltraining mit DP im Vergleich zum Modelltraining ohne DP erhöht.
✓	**HT_1.4**	Die Anzahl an Epochen steigt mit dem Privatsphären Budget.
		Datenqualität
✓	**HT_2.1**	Modelltraining ohne DP führt im Vergleich zum Modelltraining mit DP zu einer verbesserten Datenqualität.
(✗)	**HT_2.2**	Je größer das Privatsphären Budget, desto besser die Datenqualität.
✗	**HT_2.3**	Je komplexer der Datensatz, desto schlechter die Datenqualität.
✓	**HT_2.4**	Kategoriale Spalten können im Vergleich zu kontinuierlichen Spalten besser abgebildet werden.
✓	**HT_2.5**	Je geringer die Anzahl an unterschiedlichen Kategorien einer Spalte, desto besser die Datenqualität.
✓	**HT_2.6**	Je schwächer die Korrelation zwischen zwei Spalten, desto besser die Datenqualität.
		Privatsphärenschutz
✓	**HT_3.1**	Modelltraining mit DP führt im Vergleich zum Modelltraining ohne DP zu einem erhöhten Datenschutz.
(✗)	**HT_3.2**	Je niedriger das Privatsphären Budget, desto höher der Datenschutz.
✓	**HT_3.3**	Je mehr Informationen einem Angreifer zur Verfügung stehen, desto höher das Risiko, dass unbekannte Spaltenausprägungen ermittelt werden können.

Tabelle 5.4: Abgleich der Hypothesen

Ergänzend zum Abgleich der Hypothesen lassen sich weitere interessante Erkenntnisse für die Beantwortung der Forschungsfrage ableiten:

E_1.1 Unabhängig von der Größe des Privatsphären Budgets bleibt die Trainingsdauer pro Epoche konstant (Ausnahme: Modell DP-CGANS).

E_1.2 Die Anstiegsgröße des Privatsphären Budgets je Epoche verringert sich im Trainingsverlauf, was zu einer Verstärkung von **HT_1.4** führt.

E_2.1 Es bestehen große Qualitätsunterschiede zwischen den DP-Ursprungsmodellen (DPGAN & PATE) und den fortgeschrittenen Modellen (CTAB-GAN+ & DP-CGANS).

E_2.2 Die Beschaffenheit einzelner Spalten lässt sich besser erlernen als die Korrelation zwischen Spalten.

E_2.3 Korrelationen zwischen zwei numerischen Spalten werden besser erlernt als Korrelationen zwischen unterschiedlichen bzw. kategorischen Spalten.

E_2.4 Das Abbilden der Korrelation zwischen einer kategorischen und numerischen Spalte fällt den Modellen besonders schwer.

E_2.5 Es werden ausschließlich neue Datensätze generiert (keine Kopien).

E_2.6 DPGAN & PATE-GAN reproduzieren die numerischen Spalten nicht geeignet. Auch wenn sie die Wertebereiche weitgehend abdecken, ergeben die Daten ohne Einhaltung der Grenzwerte keinen Sinn.

E_3.1 Die Risiken einer Verknüpfbarkeit von Daten sind begrenzt (kleiner 1%) und liegen unterhalb der anderen durch Angriffe entstehenden Risiken.

E_3.2 Die Risiken für eine Identifikation bei Angriffen mit zusammenhängenden Spalten (Multi-Variante) sind hoch. Beim Training ohne Differential Privacy erreichen sie Werte über 20%.

E_3.3 Kategoriale Spalten, bei denen eine Ausprägung den Großteil der Vorkommen ausmacht, bergen ein vergleichsweise hohes Risiko der Inferenz.

6 Diskussion

Folgend werden der Nutzen der Integration von DP in GANs diskutiert sowie die nicht eingetroffenen Hypothesen **HT_2.2** und **HT_3.2** (Privatsphären Budget beeinflusst Datenqualität und Datenschutz) untersucht. Ferner werden Verbesserungspotentiale der Evaluation aufgezeigt sowie die Forschungsfrage anhand der Ergebnisse der Teilfragen beantwortet. Abschießend erfolgt die Modellauswahl für DaFne einschließlich eines Anforderungsabgleichs sowie einer kurzen Erläuterung zum Quellcode.

6.1 Anwendbarkeit von DP in GANs

Die Ergebnisse aus Kapitel 5 demonstrieren den Einfluss der Integration von Differential Privacy in GANs. Es zeigt sich, dass DP erfolgreich für einen verbesserten Schutz der Privatsphäre genutzt werden kann. Fast in allen Fällen besitzen die mit Differential Privacy generierten Daten ein geringeres Sicherheitsrisiko als die Daten, die ohne DP trainiert wurden (**HT_3.1**). Gleichermaßen wird sichtbar, dass die mit DP generierten Daten eine vergleichsweise geringere Qualität aufweisen (**HT_2.1**). Der in Kapitel 2.3.4 vorgestellte Trade-Off zwischen Nutzbarkeit und Privatsphäre wird ersichtlich.

Obgleich die Reduzierung von Risiken allgemein (DP vs. kein DP) belegt werden kann, können die Hypothesen zur Auswirkung der Größe des Privatsphären Budgets auf Datenqualität (**HT_2.2**) und Datenschutz (**HT_3.2**) nicht vollständig bestätigt werden. Da dieses Verhalten aufgrund der kürzeren Trainingsdauer wenig sinnvoll erscheint, bedarf es einer Analyse möglicher Ursachen dieses Verhaltens.

Beim Fokus auf die Fälle der fortgeschrittenen Modelle (CTAB-GAN+ & DP-CGANS), bei denen die Hypothese **HT_2.2** nicht zu trifft, wird deutlich, dass vor allem Daten mit einem Privatsphären Budget von drei im Vergleich zu zehn besser abschneiden. Die zwei Metriken KS Complement (Beschaffenheit von numerischen Daten) und Contigency Similarity (Korrelationen zwischen unterschiedlichen Datentypen) tragen hierfür die

Verantwortung. Die Ergebnisse zeigen, dass diese beiden Metriken Eigenschaften untersuchen, die für die Modelle schwierig zu erlernen sind (vgl. **HT_2.4** & **E_2.4**). Darüber hinaus fällt auf, dass bei der Generierung der Energie Daten die Hypothese **HT_2.2** öfter nicht zutrifft. Es handelt sich auch hierbei um den Datensatz, der von den Modellen schwerer zu erlernen ist (vgl. **HT_2.3**).

Im Gegensatz zur Qualität wird die Hypothese **HT_3.2** beim Energie Datensatz im Vergleich zum AGMA Datensatz häufiger eingehalten. Lediglich bei den Angriffen zur Inferenz sowie bei der Multi-Variante zur Identifizierung mit fünf verknüpften Spalten übersteigen die Daten mit einem Privatsphären Budget von zehn die Risiken der Daten mit einem Privatsphären Budget von 50. Bei den AGMA Daten fallen vergleichsweise zu hohe Risiken bei den Datensätzen mit einem Privatsphären Budget von 3 sowie 10 auf. Unmittelbare Auswirkungen von Qualität auf Risiken können nicht identifiziert werden.

Um fundierte Gründe für die unerwarteten Ergebnisse der Hypothesen **HT_2.2** und **HT_3.2** liefern zu können, werden zusätzliche Analysen zur Datenqualität und Sicherheit in weiterführenden Forschungsarbeiten notwendig. Bei diesen sollte auch über eine Erweiterung der Experimente dieser Thesis nachgedacht werden. Dadurch dass die Modelle weder während des Trainings noch bei der direkten Synthese deterministisch sind, sollten die Datensätze sowie die zugehörigen Modelle nicht nur einmal generiert bzw. trainiert werden. Zukünftige Evaluationen sollten daher auf mehrmals trainierten Modellen mit mehrfach generierten Daten aufbauen.

Für eine zuverlässige Vergleichbarkeit von Datensätzen mit unterschiedlichen Privatsphären Budgets sollten zukünftig zudem alle Datensätze mit Privatsphären Budget innerhalb eines Modelltrainings generiert werden. Konkret werden dann Daten nach dem Erreichen des jeweiligen Privatsphären Budgets generiert und nicht das Modelltraining jeweils neu gestartet. Des Weiteren sollten zusätzlich zu den Evaluationen mit SDMetrics und Anonymeter ergänzende Metriken bei der Validierung der Qualität und des Privatsphärenschutzes unterstützen.

6.2 Beantwortung der Forschungsfrage

Auch wenn die Evaluation Verbesserungspotential aufweist, kann die Forschungsfrage mit Hilfe der zugehörigen Teilfragen eindeutig beantwortet werden:

Performance: Wie viel Zeit benötigt das Modell für die Synthese von Daten?
Bis auf eine zu vernachlässigende Ausnahme generiert das CTAB-GAN+ am schnellsten die geforderten Daten. Insbesondere mit der Zunahme des Privatsphären Budgets und der damit einhergehenden verlängerten Trainingsdauer werden große Unterschiede zwischen dem CTAB-GAN+ und den anderen drei Modellen sichtbar. Beim Training ohne DP und dem niedrigen Privatsphäre Budget von drei erzielt das DP-CGANS mit dem CTAB-GAN+ vergleichbare Zeiten. Beim DP-CGANS muss jedoch berücksichtigt werden, dass die Anzahl an Dimensionen im Datensatz einen großen Einfluss auf die Dauer besitzt und folglich der AGMA Datensatz für dieses Modell auf 25 Spalten gekürzt werden musste.

Zusammenfassend sticht das **CTAB-GAN+** bei der Modell-Performance besonders durch seine vergleichsweise kurzen Trainingszeiten bei hohen Privatsphären Budgets sowie bei der Generierung von Daten mit einer großen Anzahl an Dimensionen hervor. Die benötigten Zeiten für die unterschiedlichen Generierungen der AGMA Daten betragen bei verwendeter Parameterwahl (siehe Tabelle A.1) **4min (ϵ=3)**, **57min (ϵ=10)**, **12h 47min (ϵ=50)** und **1h 20min (ϵ=∞)**. Minimal geringer sind die Zeiten bei der Generierung der Energie Daten (vgl. Tabelle A.2).

Datenqualität: Inwiefern entsprechen die Eigenschaften der vom Modell generierten Daten denen der Trainingsdaten?
Im Allgemeinen existieren bezüglich der Qualität der generierten Daten große Unterschiede zwischen den Ursprungsmodellen sowie fortgeschrittenen Modellen. Während beim DPGAN und PATE-GAN die numerischen Spalten ungenügend reproduziert werden und sich die Gesamtqualität immer unter einem Wert von 0,5 befindet, erreicht CTAB-GAN+ in allen Fällen eine Gesamtqualität von über 0,8 und Spitzenwerte um die 0,95. Auch DP-CGANS generiert nutzbare Daten mit einer Gesamtqualität von über 0,8 unabhängig von der Datenart. Hierbei überzeugt insbesondere die Datenqualität der ohne DP generierten Daten sowie die AGMA Daten, die mit einem Privatsphären Budget von 50 generiert wurden. Dennoch unterliegen alle Teilergebnisse des DP-CGANS den Qualitätsergebnissen des CTAB-GAN+. Darüber hinaus ist die Datenqualität auch bei den vom CTAB-GAN+ generierten Daten mit einem Privatsphären Budget von 3 und 10 hoch.

Analog zur Performance erzielt das **CTAB-GAN+** ebenfalls die beste Datenqualität. Die AGMA Daten können besser abgebildet werden als die Energie Daten. Die Datenbeschaffenheit der einzelnen Spalten kann bei den AGMA Daten zu **89%**

(ϵ=3), **90% (ϵ=10)**, **92% (ϵ=50)** und **94% (ϵ=∞)** abgedeckt werden. Die Korrelationen werden etwas schlechter nachgebildet, erreichen aber auch Werte von **85% (ϵ=3)**, **87% (ϵ=10)**, **89% (ϵ=50)** und **93% (ϵ=∞)**. Mit eine Gesamtqualität von mind. **82%**, Datenbeschaffenheit von mind. **84%** sowie Korrelationswerten von mind. **80%** kann das CTAB-GAN+ die Energie Daten etwas weniger präzise abbilden. Die Ergebnisse bleiben dennoch im Vergleich zu den anderen Modellen die mit der höchsten Qualität.

Privatsphärenschutz: Wie sicher sind die vom Modell generierten Daten gegenüber Angriffen?

Die Risikobewertung für einen Verlust der Privatsphäre wurde ausschließlich auf Datensätzen mit hoher Qualität vorgenommen. Folglich existiert nur ein Vergleich zwischen den vom DP-CGANS und CTAB-GAN+ generierten Datensätzen. Es zeigt sich, dass die Höhe der Risiken sowie dessen Minimierung durch die Integration von DP im Wesentlichen von der Angriffsart abhängt. Während sich die Risiken einer Verknüpfbarkeit als gering herausstellen, entstehen höhere Risiken bei der Identifizierung (vor allem bei der Multi-Variante) sowie bei der Inferenz einzelner Spalten. Auffällig sind die vergleichsweise hohen Risiken der Identifizierung (Uni-Variante) bei den vom DP-CGANS ohne DP generierten Daten, die bei einem Training mit Privatsphären Budget von 50 fast vollständig eliminiert werden können. Auch die Risiken der Multi-Variante können vom DP-CGANS durch den Einsatz von DP reduziert werden und unterliegen zumeist den Risiken des CTAB-GAN+. Bei den Risiken der Inferenz erzielt das CTAB-GAN+ geringere Werte beim AGMA Datensatz und das DP-CGANS geringere Werte beim Energie Datensatz.

Im Gegensatz zur präzisen Modellwahl bezüglich Performance und Qualität kann beim Privatsphären Schutz keine eindeutige Auswahl getroffen werden. Beide Modelle generieren Daten mit Risiken, die jedoch durch die Integration von DP reduziert werden können. Generell trägt die Angriffsart maßgeblich zu den unterschiedlich hohen Risiken bei. Insbesondere bei der Identifizierung bestehen hohe Risiken, die bei der Multi-Variante mit fünf Spalten beim AGMA Datensatz ihre Höchstwerte erreichen. **CTAB-GAN+** senkt die Risiken **von 23%** (Training ohne DP) **auf 17%** (Training mit ϵ=50) und **DP-CGANS von 20% auf 15%**. Ergänzend sollte betont werden, dass der Vergleich von zwei Datensätzen keine abschließende Entscheidung über das sicherere Modell zulässt. Weitere Experimente werden nötig.

Forschungsfrage: Welches Generative Adversarial Network eignet sich für eine adäquate Synthese sensibler tabellarischer Daten unter Berücksichtigung von Differential Privacy?

Auf Grundlage der Ergebnisse der Teilfragen fällt die Wahl für ein geeignetes GAN mit Integration von DP auf das **CTAB-GAN+**. Das Modell überzeugt sowohl bei der Trainingsdauer als auch bei der generierten Datenqualität und gewährleistet einen verbesserten Privatsphärenschutz durch die Integration von DP. Vor allem aufgrund der guten Datenqualität trotz geringem Privatsphären Budget und der unproblematischen Generierung von hoch-dimensionalen Daten kann sich das CTAB-GAN+ gegenüber den anderen untersuchten Modellen durchsetzen. Nichtsdestotrotz besitzt auch das CTAB-GAN+ Verbesserungspotentiale, die in fortführenden Arbeiten Berücksichtigung finden sollten. Indizien geben hierfür beispielsweise die gewonnen Erkenntnisse **E_2.2** bis **E_2.4** und Hypothesen wie **HT_2.4** bis **HT_2.6** (siehe Kapitel 5.4).

6.3 Modellauswahl für DaFne

Durch die im Zuge dieser Arbeit durchgeführten Experimente, fällt die Modellauswahl für die DaFne Plattform auf das CTAB-GAN+. Wie aus den Ergebnissen und beantworteten Forschungsfragen hervorgeht, setzt es sich insbesondere durch eine kurze Trainingsdauer und der Generierung von qualitativ hochwertigen Daten durch. Darüber hinaus kann der positive Einfluss der Integration von DP auf den Privatsphärenschutz belegt werden. Tabelle 6.1 gibt einen Überblick der umgesetzten funktionalen Anforderungen (definiert in Kapitel 4.1.1). Die offenen Anforderungen weisen auf Handlungspotentiale für weiterführende Arbeiten hin.

Erfüllt	Id	Beschreibung
		Modellaufbau
✓	**FA_01**	Modell garantiert beim Training Privatsphäre
✓	**FA_02**	Architektur des Modells ist skizziert
✓	**FA_03**	Modelltraining, Vor- und Nachbearbeitung sind nachvollziehbar
✗	**FA_04**	Trainiertes Modell bleibt für erneute Datensynthese gespeichert

		Modelltraining
✓	**FA_05**	Beispieldatensatz steht zur Verfügung
✓	**FA_06**	Trainingsparameter sind frei wählbar
✓	**FA_07**	Hilfestellung bei der Wahl der Trainingsparameter ist vorhanden
✓	**FA_08**	Default Trainingsparameter werden angezeigt
✗	**FA_09**	Trainingsdauer und benötigte Epochenanzahl werden prognostiziert
(✗)	**FA_10**	Trainingsfortschritt der Generierung wird angezeigt
✗	**FA_11**	Sobald die Daten generiert sind erfolgt eine Benachrichtigung
		Generierte Daten
✓	**FA_12**	Unabhängig von Eigenschaften und Domäne sind Trainingsdaten wählbar
✓	**FA_13**	Generierte Daten lassen keinen Rückschluss auf reale Daten zu
✓	**FA_14**	Generierte Daten besitzen die Eigenschaften der realen Daten
✓	**FA_15**	Metriken zur Überprüfung der einzuhaltenden Privatsphäre sind verfügbar
✓	**FA_16**	Metriken zur Qualitätsüberprüfung sind vorhanden
✗	**FA_17**	Generierte Daten sind in gleicher Weise geeignet für KI-Anwendungen

Tabelle 6.1: Abgleich der funktionalen Anforderungen

Die nicht-funktionalen Anforderungen betreffen nicht nur das untersuchte Private Modell, sondern die gesamte Plattform. Spezifisch kann festgehalten werden, dass beim CTAB-GAN+ sowohl vorgegebene Parameter als Default verwendet als auch von erfahrenen Nutzern modifiziert werden können **(NFA_01)**. Des Weiteren überzeugt das Modell durch seine hohe Perfomance bezüglich generierter Qualität und Trainingsdauer **(NFA_02)**. Seine Architektur sowie gewählten Parameterwerte werden in dieser Thesis vorgestellt und können in einer ausführlichen Dokumentation für die Plattform bereitgestellt werden **(NFA_05)**.

Während der Generator des GANs selbst und auch die generierten Daten einen erhöhten Privatsphärenschutz garantieren, muss auch die Plattform im Allgemeinen vor Angriffen geschützt werden **(NFA_06)**. Zudem sind **Zuverlässigkeit (NFA_03)** und **Erweiterbarkeit (NFA_04)** weitere wichtige Anforderungen die ferner bei der Entwicklung der Plattform Berücksichtigung finden sollten.

In Ergänzung zu der Vorstellung der Architektur des CTAB-GAN+ (Kapitel 3.1) und der Beschreibung sowie Wahl der Parameter (Kapitel 4.5 & Tabelle A.1) befindet sich im Anhang A.2 der für die Experimente modifizierte Quellcode. Anhand des häufig verwendeten Beispieldatensatzes „Adult Income“ können Nutzer erste private Daten generieren. Als **Input** benötigt das Modell folgende Informationen:

1. Realer Datensatz (CSV-Datei)
2. Metadaten mit Informationen zur Datenart der einzelnen Spalten und die ausgewählte Spalte für die zusätzliche Komponente C (JSON-Datei)
3. Integration von DP (Boolean):
   a) True: Privatsphären Budget (Integer)
   b) False: Anzahl an zu trainierenden Epochen (Integer)

Zusätzlich zur Terminalausgabe des Modellfortschritts werden Zeit, Anzahl der Epoche sowie Privatsphären Budget in einer CSV-Datei für eine anschließende Evaluation der Performance gespeichert. Die generierten Daten befinden sich nach der Synthese im Ordner: Fake_Datasets.

Ein für die Plattform ebenfalls relevanter Nutzen, jedoch nicht Hauptbestandteil dieser Arbeit, liegt in der Analyse und Bereitstellung geeigneter Evaluations-Metriken. Die in der Evaluation angewendeten Skripte (Anonymeter) und Metriken (SDMetriks) sind der Thesis ebenfalls angehängt (siehe Anhang A.1 & A.2). Sie können als Ausgangspunkt für Projekte dienen, die sich auf die Evaluation der erzeugten Daten konzentrieren, insbesondere im Hinblick auf Datenschutz.

7 Zusammenfassung

Abschließend werden in diesem Kapitel die zentralen Elemente der Thesis zusammengefasst und um einen Ausblick auf nachfolgende Projekte ergänzt.

Zielsetzung Mit dem Ziel schützenswerte Daten ohne Verlust von Privatsphäre nutzbar zu machen, beschäftigt sich die Thesis mit der Integration von Differential Privacy in Generative Adversarial Networks. Im Rahmen des Forschungsprojektes DaFne wird konkret nach einem geeigneten DP-GAN gesucht, das zum einen die Eigenschaften der realen Daten abbildet (1) und zum anderen die Privatsphäre schützt (2). Darüber hinaus beeinflusst die Modellperformance, insbesondere die Trainingsdauer, die Verwendung des Modells (3).

Aufbau der Experimente Auf Grundlage der drei Anforderungen erfolgt der Aufbau der Experimente. Nach der Untersuchung von Trainingszeit und Eigenschaften einzelner Epochen werden Qualitäts-Metriken auf die generierten Daten angewendet sowie die Risiken unterschiedlicher Angriffsformen berechnet. Die Evaluation basiert auf einem realen sowie einem simulierten Datensatz, die jeweils von vier unterschiedlichen DP-GANs reproduziert werden. Da die Größe des Privatsphären Budgets Einfluss auf Qualität und Privatsphärenschutz besitzen kann, werden zusätzlich zum allgemeinen Vergleich vom Training mit und ohne DP explizit Daten mit ϵ-Werten von drei, zehn und fünfzig generiert. Aufgrund der Tatsache, dass sich nicht alle Modelle für ein Training ohne DP eignen sowie einzelne Modelle zu viel Trainingszeit beanspruchen, ergeben sich insgesamt 27 zu evaluierende Datensätze.

Evaluationsergebnisse Die verwendeten Modelle unterscheiden sich einerseits in der Art der Integration von Differential Privacy in GANs (DP-SGD vs. PATE), andererseits in ihrem Entwicklungsstand. Die Ursprungsmodelle DPGAN & PATE-GAN stehen den fortgeschrittenen Modellen CTAB-GAN+ & DP-CGANS gegenüber. Vor allem innerhalb der Qualitätskontrolle der Daten werden große Unterschiede sichtbar. Die sich über die letzten Jahre weiterentwickelten Vor- & Nachbearbeitungsschritte tabellarischer Datenverarbeitung (verwendet in CTAB-GAN+ & DP-CGANS) tragen u.a. zu einer erhöhten Datenqualität bei.

Beim Vergleich der Modelle zeigt sich zudem, dass die Eigenschaften der Datensätze einen großen Einfluss auf die Modellleistung haben. Während das DP-CGANS Schwierigkeiten bei der Reproduktion hoch-dimensionaler Daten besitzt, können DPGAN und PATE-GAN numerische Daten nicht geeignet abbilden. Grundsätzlich fällt auf, dass alle vier Modelle kategoriale Daten besser reproduzieren können als numerische und die Datenbeschaffenheit einzelner Spalten gegenüber Korrelationen zwischen Spalten verbessert abgebildet wird. Außerdem beeinflussen die Anzahl unterschiedlicher Kategorien sowie die Korrelationsstärke zwischen Spalten die generierte Datenqualität. Die Resultate der Risikoberechnungen für einen Verlust der Privatsphäre demonstrieren, dass die Höhe der Risiken sowie dessen Minimierung im Wesentlichen von der Angriffsart abhängt. Im Gegensatz zur Verknüpfbarkeit, bei der die Risiken unter 1% liegen, erreichen die Risiken bezüglich der Identifikation Werte über 20%.

Insgesamt belegen die Evaluationsergebnisse, dass die Integration von Differential Privacy in GANs einen erhöhten Privatsphärenschutz ermöglicht. Auf der anderen Seite offenbart sich aber auch der Trade-Off zwischen Nutzbarkeit und Datenschutz. Der gesteigerte Schutz geht mit einer Beeinträchtigung der Datenqualität einher. Darüber hinaus erhöht sich die Trainingsdauer durch die Integration von DP. Entgegen den Erwartungen kann innerhalb der durchgeführten Experimente nicht vollständig bewiesen werden, dass sich mit Zunahme des Privatsphären Budgets die Datenqualität verbessert und der Datenschutz sinkt. Weitere Evaluationen werden benötigt, um zuverlässige Aussagen zu den Ursachen dieses Verhaltens ableiten zu können.

Beantwortung der Forschungsfrage Final kann die Forschungsfrage hinsichtlich der Auswahl eines passenden GANs unter Einbeziehung von DP durch die Auswertung der einzelnen Aspekte beantwortet werden. Das Modell CTAB-GAN+ überzeugt bezüglich Trainingsdauer sowie Datenqualität und erzielt einen erhöhten Privatsphärenschutz durch die Integration von DP. Insbesondere übertrifft es die anderen Modelle durch eine hohe Datenqualität auch bei geringem Privatsphären Budget sowie durch seine Leistung bei der Generierung hoch-dimensionaler Daten.

Verbesserungspotential Es ist wichtig zu betonen, dass die Evaluationsergebnisse ausschließlich auf den durchgeführten Experimenten beruhen und auf den Rahmen dieser Untersuchung beschränkt sind. Zusätzlich zu einer erweiterten Evaluation besteht Verbesserungspotential bezüglich der Wiederholungen von Modelltraining und Datengenerierung. Aufgrund des nicht deterministischen Charakters beim Training der Modelle sowie bei der Generierung der Daten, sollten Daten unter der gleichen Bedingung (Modell & Datensatz & Privatsphären Budget) mehrfach generiert werden. Darüber hinaus fehlt in den Experimenten eine Überwachung zum Overfitting der Modelle. Vor allem beim Training ohne DP besteht das Risiko einer zu exakten Anpassung, weshalb für die Generierung bestmöglicher Daten auch Evaluationen von Zwischenergebnissen erforderlich werden.

Des Weiteren sollten die Hypothesen zum Einfluss der Größe des Privatsphären Budgets (HT_2.2 & HT_3.2) erneut überprüft werden. Für eine zuverlässige Vergleichbarkeit von Datensätzen mit unterschiedlichem Privatsphären Budget sollten zukünftig alle Datensätze mit Privatsphären Budget innerhalb eines Modelltrainings generiert werden. Konkret werden dann Daten nach dem Erreichen des jeweiligen Privatsphären Budgets generiert und nicht das Modelltraining jeweils neu gestartet.

Ausblick Im Rahmen des Forschungsprojekts DaFne sind die nächsten Schritte darauf ausgerichtet, die verbleibenden Anforderungen zu erfüllen. Explizit muss das trainierte Modell für eine erneute Datensynthese gespeichert und Prognosen sowie Anzeigen zum Trainingsfortschritt integriert werden. Darüber hinaus ist zu überprüfen, ob sich die generierten Daten genauso gut wie die realen Daten für KI-Anwendungen eignen. Abschließend sollte das CTAB-GAN+ in einem Docker-Container bereitgestellt werden, um Modularität für Erweiterbarkeit, Wiederverwendung und Wartbarkeit zu gewährleisten. In Ergänzung zur Bereitstellung des eigentlichen Modells können die in der Evaluation verwendeten Metriken als Einstiegspunkt für eine Analyse und Auswahl geeigneter Evaluations-Metriken u.a. mit Fokus auf Datenschutz für die Plattform genutzt werden.

Generell können die Evaluationsergebnisse nicht nur zur Wahl eines geeigneten Modells beitragen, sondern weisen auch auf Schwachstellen und Stärken der Modelle hin. Insbesondere die schwieriger abzubildenden Eigenschaften der Datensätze (wie z.B. kontinuierliche Daten oder Korrelationen zwischen Spalten) können auf Optimierungspotentiale in den Modellen hinweisen, die Fokus weiterführender Arbeiten sein könnten.

Im Bezug auf den Trade-Off zwischen Qualität und Privatsphäre bleibt die Frage idealer Parametergrößen offen. In Abhängigkeit vom Zweck, Datensatz und Modell sollten Empfehlungen zur Initialisierung von ϵ & δ gegeben werden. Auch der Vorschlag einer individualisierten Differential Privacy von Boensich et. al (siehe Kapitel 3.2) könnte weiterverfolgt und in den Modellen zur Reduzierung des Trade-Offs integriert werden. Ferner könnte der Differential Privacy Ansatz alternativen Varianten zur Sicherstellung von Privatsphäre gegenübergestellt werden.

Literaturverzeichnis

[1] Abadi, Martín ; Andersen, David G.: Learning to Protect Communications with Adversarial Neural Cryptography. In: *arXiv preprint arXiv:1610.06918* (2016)

[2] Abadi, Martin ; Chu, Andy ; Goodfellow, Ian ; McMahan, H. B. ; Mironov, Ilya ; Talwar, Kunal ; Zhang, Li: Deep Learning with Differential Privacy. In: *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security* (2016), Oktober, S. 308–318. ISBN 9781450341394

[3] Arbeitsgemeinschaft Media-Analyse e.V.: *Website: AGMA*. https://www.agma-mmc.de/ueber/agma. – Zugriffsdatum: 2023-09-27

[4] Arjovsky, Martin ; Chintala, Soumith ; Bottou, Léon: Wasserstein Generative Adversarial Networks. In: *Proceedings of the 34th International Conference on Machine Learning*, PMLR, Juli 2017, S. 214–223. – ISSN 2640-3498

[5] Armanious, Karim ; Jiang, Chenming ; Fischer, Marc ; Küstner, Thomas ; Hepp, Tobias ; Nikolaou, Konstantin ; Gatidis, Sergios ; Yang, Bin: MedGAN: Medical Image Translation Using GANs. In: *Computerized Medical Imaging and Graphics* 79 (2020), Januar, S. 101684. – ISSN 0895-6111

[6] Augenstein, Sean ; McMahan, H. B. ; Ramage, Daniel ; Ramaswamy, Swaroop ; Kairouz, Peter ; Chen, Mingqing ; Mathews, Rajiv: Generative Models for Effective ML on Private, Decentralized Datasets. In: *arXiv preprint arXiv:1911.06679* (2019)

[7] Beaulieu-Jones, Brett K. ; Wu, Zhiwei S. ; Williams, Chris ; Lee, Ran ; Bhavnani, Sanjeev P. ; Byrd, James B. ; Greene, Casey S.: Privacy-Preserving Generative Deep Neural Networks Support Clinical Data Sharing. In: *Circulation: Cardiovascular Quality and Outcomes* 12 (2019), Nr. 7, S. e005122

[8] Bengio, Yoshua ; Lecun, Yann ; Hinton, Geoffrey: Deep Learning for AI. In: *Communications of the ACM* 64 (2021), Nr. 7, S. 58–65

[9] Bernau, Daniel: *Improved Usability of Differential Privacy in Machine Learning : Techniques for Quantifying the Privacy-Accuracy Trade-Off*, doctoralThesis, 2022

[10] Boenisch, Franziska: Privatsphäre und Maschinelles Lernen. In: *Datenschutz und Datensicherheit - DuD* 45 (2021), Juli, Nr. 7, S. 448–452. – ISSN 1862-2607

[11] Boenisch, Franziska ; Mühl, Christopher ; Dziedzic, Adam ; Rinberg, Roy ; Papernot, Nicolas: *Have It Your Way: Individualized Privacy Assignment for DP-SGD*. März 2023

[12] Boenisch, Franziska ; Mühl, Christopher ; Rinberg, Roy ; Ihrig, Jannis ; Dziedzic, Adam: Individualized PATE: Differentially Private Machine Learning with Individual Privacy Guarantees. In: *Proceedings on Privacy Enhancing Technologies* 1 (2023), S. 158–176

[13] Borealis AI: *Private Data Generation Toolbox*. https://github.com/BorealisAI/private-data-generation. November 2023. – Zugriffsdatum: 2023-11-07

[14] Bundesamt für Sicherheit in der Informationstechnik: *Smart City*. https://www.bsi.bund.de/DE/Themen/Unternehmen-und-Organisationen/Informationen-und-Empfehlungen/Smart-City/smart-city.html?nn=1006472. – Zugriffsdatum: 2023-09-28

[15] Chen, Dingfan ; Orekondy, Tribhuvanesh ; Fritz, Mario: Gs-Wgan: A Gradient-Sanitized Approach for Learning Differentially Private Generators. In: *Advances in Neural Information Processing Systems* 33 (2020), S. 12673–12684

[16] Choi, Edward ; Biswal, Siddharth ; Malin, Bradley ; Duke, Jon ; Stewart, Walter F. ; Sun, Jimeng: Generating Multi-label Discrete Patient Records Using Generative Adversarial Networks. In: *Proceedings of the 2nd Machine Learning for Healthcare Conference*, PMLR, November 2017, S. 286–305. – ISSN 2640-3498

[17] Creswell, Antonia ; White, Tom ; Dumoulin, Vincent ; Arulkumaran, Kai ; Sengupta, Biswa ; Bharath, Anil A.: Generative Adversarial Networks: An Overview. In: *IEEE Signal Processing Magazine* 35 (2018), Januar, Nr. 1, S. 53–65. – ISSN 1558-0792

[18] DataCebo: *SDMetrics*. https://docs.sdv.dev/sdmetrics/. – Zugriffsdatum: 2013-11-04

[19] Donahue, Jeff ; Krähenbühl, Philipp ; Darrell, Trevor: Adversarial Feature Learning. In: *arXiv preprint arXiv:1605.09782* (2016)

[20] Dumoulin, Vincent ; Belghazi, Ishmael ; Poole, Ben ; Mastropietro, Olivier ; Lamb, Alex ; Arjovsky, Martin ; Courville, Aaron: Adversarially Learned Inference. In: *arXiv preprint arXiv:1606.00704* (2016)

[21] Durugkar, Ishan ; Gemp, Ian ; Mahadevan, Sridhar: Generative Multi-Adversarial Networks. In: *arXiv preprint arXiv:1611.01673* (2016)

[22] Dwork, Cynthia: Differential Privacy. In: *International Colloquium on Automata, Languages, and Programming*, Springer, 2006, S. 1–12

[23] Dwork, Cynthia ; Roth, Aaron: The Algorithmic Foundations of Differential Privacy. In: *Foundations and Trends® in Theoretical Computer Science* 9 (2013), Nr. 3-4, S. 211–407. – ISSN 1551-305X, 1551-3068

[24] Esteban, Cristóbal ; Hyland, Stephanie L. ; Rätsch, Gunnar: Real-Valued (Medical) Time Series Generation with Recurrent Conditional Gans. In: *arXiv preprint arXiv:1706.02633* (2017)

[25] Fan, Ju ; Liu, Tongyu ; Li, Guoliang ; Chen, Junyou ; Shen, Yuwei ; Du, Xiaoyong: Relational Data Synthesis Using Generative Adversarial Networks: A Design Space Exploration. In: *arXiv preprint arXiv:2008.12763* (2020)

[26] Fan, Liyue: A Survey of Differentially Private Generative Adversarial Networks. In: *The AAAI Workshop on Privacy-Preserving Artificial Intelligence*, 2020, S. 8

[27] Foster, David: *Generative deep learning: teaching machines to paint, write, compose, and play.* First edition. Sebastopol, CA : O'Reilly Media, Inc, 2019. – ISBN 978-1-4920-4194-8

[28] Frigerio, Lorenzo ; de Oliveira, Anderson S. ; Gomez, Laurent ; Duverger, Patrick: Differentially Private Generative Adversarial Networks for Time Series, Continuous, and Discrete Open Data. In: *ICT Systems Security and Privacy Protection: 34th IFIP TC 11 International Conference, SEC 2019, Lisbon, Portugal, June 25-27, 2019, Proceedings 34*, Springer, 2019, S. 151–164

[29] Gao, Hongchang ; Pei, Jian ; Huang, Heng: Progan: Network Embedding via Proximity Generative Adversarial Network. In: *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2019, S. 1308–1316

[30] Ghosh, Arnab ; Kulharia, Viveka ; Namboodiri, Vinay P. ; Torr, Philip H. ; Dokania, Puneet K.: Multi-Agent Diverse Generative Adversarial Networks. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, S. 8513–8521

[31] Giomi, Matteo ; Boenisch, Franziska ; Wehmeyer, Christoph ; Tasnádi, Borbála: A Unified Framework for Quantifying Privacy Risk in Synthetic Data. In: *arXiv preprint arXiv:2211.10459* (2022), November

[32] Goodfellow, Ian: Nips 2016 Tutorial: Generative Adversarial Networks. In: *arXiv preprint arXiv:1701.00160* (2016)

[33] Goodfellow, Ian ; Bengio, Yoshua ; Courville, Aaron: *Deep Learning*. MIT press, 2016

[34] Goodfellow, Ian ; Pouget-Abadie, Jean ; Mirza, Mehdi ; Xu, Bing ; Warde-Farley, David ; Ozair, Sherjil ; Courville, Aaron ; Bengio, Yoshua: Generative Adversarial Nets. In: *Advances in Neural Information Processing Systems* Bd. 27, Curran Associates, Inc., 2014

[35] Gui, Jie ; Sun, Zhenan ; Wen, Yonggang ; Tao, Dacheng ; Ye, Jieping: A Review on Generative Adversarial Networks: Algorithms, Theory, and Applications. In: *IEEE Transactions on Knowledge and Data Engineering* 35 (2023), April, Nr. 4, S. 3313–3332. – ISSN 1558-2191

[36] Guimaraes, Gabriel L. ; Sanchez-Lengeling, Benjamin ; Outeiral, Carlos ; Farias, Pedro Luis C. ; Aspuru-Guzik, Alán: Objective-Reinforced Generative Adversarial Networks (Organ) for Sequence Generation Models. In: *arXiv preprint arXiv:1705.10843* (2017)

[37] Gulrajani, Ishaan ; Ahmed, Faruk ; Arjovsky, Martin ; Dumoulin, Vincent ; Courville, Aaron C.: Improved Training of Wasserstein GANs. In: *Advances in Neural Information Processing Systems* Bd. 30, Curran Associates, Inc., 2017

[38] Gurumurthy, Swaminathan ; Kiran Sarvadevabhatla, Ravi ; Venkatesh Babu, R.: Deligan: Generative Adversarial Networks for Diverse and Limited Data. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, S. 166–174

[39] Hartmann, Kay G. ; Schirrmeister, Robin T. ; Ball, Tonio: EEG-GAN: Generative Adversarial Networks for Electroencephalograhic (EEG) Brain Signals. In: *arXiv preprint arXiv:1806.01875* (2018)

[40] Hong, Yongjun ; Hwang, Uiwon ; Yoo, Jaeyoon ; Yoon, Sungroh: How Generative Adversarial Networks and Their Variants Work: An Overview. In: *ACM Computing Surveys* 52 (2019), Februar, Nr. 1, S. 10:1–10:43. – ISSN 0360-0300

[41] Hradec, Jiri ; Craglia, Massimo ; Di Leo, Margherita ; De Nigris, Sarah ; Ostlaender, Nicole ; Nicholson, Nicholas: Multipurpose Synthetic Population for Policy Applications. In: *No. JRC128595* (2022)

[42] Hu, Weiwei ; Tan, Ying: Generating Adversarial Malware Examples for Black-Box Attacks Based on GAN. In: *International Conference on Data Mining and Big Data*, Springer, 2022, S. 409–423

[43] Huang, Rui ; Zhang, Shu ; Li, Tianyu ; He, Ran: Beyond Face Rotation: Global and Local Perception Gan for Photorealistic and Identity Preserving Frontal View Synthesis. In: *Proceedings of the IEEE International Conference on Computer Vision*, 2017, S. 2439–2448

[44] IfD Allensbach: *Anteil der Bevölkerung in Deutschland nach Wohnsituation und sozioökonomischem Status 2023.* https://de.statista.com/statistik/daten/studie/1313981/umfrage/wohnsituation-der-bevoelkerung-nach-soziooekonomischem-status/. Juni 2023. – Zugriffsdatum: 2023-09-28

[45] Jabbar, Abdul ; Li, Xi ; Omar, Bourahla: A Survey on Generative Adversarial Networks: Variants, Applications, and Training. In: *ACM Computing Surveys* 54 (2021), Nr. 8, S. 1–49. – ISSN 0360-0300, 1557-7341

[46] Jordon, James ; Yoon, Jinsung ; Van Der Schaar, Mihaela: PATE-GAN: Generating Synthetic Data with Differential Privacy Guarantees. In: *International Conference on Learning Representations*, 2019

[47] Karras, Tero ; Aila, Timo ; Laine, Samuli ; Lehtinen, Jaakko: Progressive Growing of Gans for Improved Quality, Stability, and Variation. In: *arXiv preprint arXiv:1710.10196* (2017)

[48] Killoran, Nathan ; Lee, Leo J. ; Delong, Andrew ; Duvenaud, David ; Frey, Brendan J.: Generating and Designing DNA with Deep Generative Models. In: *arXiv preprint arXiv:1712.06148* (2017)

[49] Kim, Youngjin ; Kim, Minjung ; Kim, Gunhee: Memorization Precedes Generation: Learning Unsupervised Gans with Memory Networks. In: *arXiv preprint arXiv:1803.01500* (2018)

[50] Kossen, Tabea ; Hirzel, Manuel A. ; Madai, Vince I. ; Boenisch, Franziska ; Hennemuth, Anja ; Hildebrand, Kristian ; Pokutta, Sebastian ; Sharma, Kartikey ; Hilbert, Adam ; Sobesky, Jan ; Galinovic, Ivana ; Khalil, Ahmed A. ; Fiebach, Jochen B. ; Frey, Dietmar: Toward Sharing Brain Images: Differentially Private TOF-MRA Images With Segmentation Labels Using Generative Adversarial Networks. In: *Frontiers in Artificial Intelligence* 5 (2022), Mai, S. 813842. – ISSN 2624-8212

[51] Kunert, Pamela ; Krause, Tom ; Zukunft, Olaf ; Steffens, Ulrike: A Platform Providing Machine Learning Algorithms for Data Generation and Fusion - An Architectural Approach. (2022)

[52] Lin, Kevin ; Li, Dianqi ; He, Xiaodong ; Zhang, Zhengyou ; Sun, Ming-Ting: Adversarial Ranking for Language Generation. In: *Advances in neural information processing systems* 30 (2017)

[53] Lin, Zinan ; Khetan, Ashish ; Fanti, Giulia ; Oh, Sewoong: Pacgan: The Power of Two Samples in Generative Adversarial Networks. In: *Advances in neural information processing systems* 31 (2018)

[54] Long, Yunhui ; Lin, Suxin ; Yang, Zhuolin ; Gunter, Carl A. ; Li, Bo: Scalable Differentially Private Generative Student Model via Pate. In: *arXiv preprint arXiv:1906.09338* (2019)

[55] Machanavajjhala, Ashwin ; Kifer, Daniel ; Gehrke, Johannes ; Venkitasubramaniam, Muthuramakrishnan: L-Diversity: Privacy beyond k-Anonymity. In: *ACM Transactions on Knowledge Discovery from Data (TKDD)* 1 (2007), Nr. 1, S. 3–es

[56] Mahawaga Arachchige, Pathum C. ; Bertok, Peter ; Khalil, Ibrahim ; Liu, Dongxi ; Camtepe, Seyit ; Atiquzzaman, Mohammed: Local Differential Privacy

for Deep Learning. In: *IEEE Internet of Things Journal* 7 (2020), Juli, Nr. 7, S. 5827–5842. – ISSN 2327-4662

[57] Maslej, Nestor ; Fattorini, Loredana ; Brynjolfsson, Erik ; Etchemendy, John ; Ligett, Katrina ; Lyons, Terah ; Manyika, James ; Ngo, Helen ; Niebles, Juan C. ; Parli, Vanessa ; Shoham, Yoav ; Wald, Russell ; Clark, Jack ; Perrault, Raymond: Artificial Intelligence Index Report 2023. (2023), Oktober

[58] Mironov, Ilya: Rényi Differential Privacy. In: *2017 IEEE 30th Computer Security Foundations Symposium (CSF)*, IEEE, 2017, S. 263–275

[59] Mirza, Mehdi ; Osindero, Simon: Conditional Generative Adversarial Nets. In: *arXiv preprint arXiv:1411.1784* (2014), November

[60] Mogren, Olof: C-RNN-GAN: Continuous Recurrent Neural Networks with Adversarial Training. In: *arXiv preprint arXiv:1611.09904* (2016)

[61] Narayanan, Arvind ; Shmatikov, Vitaly: Robust De-anonymization of Large Sparse Datasets. In: *2008 IEEE Symposium on Security and Privacy (Sp 2008)*. Oakland, CA, USA : IEEE, Mai 2008, S. 111–125. – ISBN 978-0-7695-3168-7

[62] Near, Joseph ; Darais, David: *Differential Privacy: Future Work & Open Challenges*. https://www.nist.gov/blogs/cybersecurity-insights/differential-privacy-future-work-open-challenges. Januar 2022. – Zugriffsdatum: 2023-09-25

[63] Papernot, Nicolas ; Abadi, Martín ; Erlingsson, Úlfar ; Goodfellow, Ian ; Talwar, Kunal: Semi-Supervised Knowledge Transfer for Deep Learning from Private Training Data. In: *arXiv preprint arXiv:1610.05755* (2016)

[64] Radford, Alec ; Metz, Luke ; Chintala, Soumith: Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks. In: *arXiv preprint arXiv:1511.06434* (2015)

[65] Sampaio, Silvio ; Sousa, Patricia R. ; Martins, Cristina ; Ferreira, Ana ; Antunes, Luís ; Cruz-Correia, Ricardo: Collecting, Processing and Secondary Using Personal and (Pseudo)Anonymized Data in Smart Cities. In: *Applied Sciences* 13 (2023), Januar, Nr. 6, S. 3830. – ISSN 2076-3417

[66] Sarker, Iqbal: AI-Based Modeling: Techniques, Applications and Research Issues Towards Automation, Intelligent and Smart Systems. In: *SN Computer Science* 3 (2022), Februar

[67] Saxena, Divya ; Cao, Jiannong: Generative Adversarial Networks (GANs): Challenges, Solutions, and Future Directions. In: *ACM Computing Surveys* 54 (2021), Mai, Nr. 3, S. 1–42. – ISSN 0360-0300, 1557-7341

[68] Schlegl, Thomas ; Seeböck, Philipp ; Waldstein, Sebastian M. ; Schmidt-Erfurth, Ursula ; Langs, Georg: Unsupervised Anomaly Detection with Generative Adversarial Networks to Guide Marker Discovery. In: *Information Processing in Medical Imaging: 25th International Conference, IPMI 2017, Boone, NC, USA, June 25-30, 2017, Proceedings*, Springer, 2017, S. 146–157

[69] Srivastava, Akash ; Valkov, Lazar ; Russell, Chris ; Gutmann, Michael U. ; Sutton, Charles: Veegan: Reducing Mode Collapse in Gans Using Implicit Variational Learning. In: *Advances in neural information processing systems* 30 (2017)

[70] Statistisches Bundesamt: *Umweltökonomische Gesamtrechnungen - Private Haushalte und Umwelt - Berichtszeitraum 2000 - 2020.* https://www.destatis.de/DE/Themen/Gesellschaft-Umwelt/Umwelt/UGR/private-haushalte/Publikationen/Downloads/haushalte-umwelt-pdf-5851319.html. 2020. – Zugriffsdatum: 2023-09-28

[71] Statistisches Bundesamt: *Stromverbrauch der privaten Haushalte nach Haushaltsgrößenklassen.* https://www.destatis.de/DE/Themen/Gesellschaft-Umwelt/Umwelt/UGR/private-haushalte/Tabellen/stromverbrauch-haushalte.html. 2021. – Zugriffsdatum: 2023-09-28

[72] Statistisches Bundesamt: *Bevölkerung nach Nationalität und Geschlecht.* https://www.destatis.de/DE/Themen/Gesellschaft-Umwelt/Bevoelkerung/Bevoelkerungsstand/Tabellen/deutsche-nichtdeutsche-bevoelkerung-nach-geschlecht-deutschland.html. 2022. – Zugriffsdatum: 2023-09-28

[73] Statistisches Bundesamt: *Altersstruktur der Bevölkerung in Deutschland 2022.* https://de.statista.com/statistik/daten/studie/1351/umfrage/altersstruktur-der-bevoelkerung-deutschlands/. Juni 2023. – Zugriffsdatum: 2023-09-28

[74] Statistisches Bundesamt: *Statistischer Bericht - Mikrozensus - Haushalte und Familien - Erstergebnisse 2022.* März 2023. – Zugriffsdatum: 2023-09-28

[75] Su, Dong ; Cao, Jianneng ; Li, Ninghui ; Bertino, Elisa ; Jin, Hongxia: Differentially Private K-Means Clustering. In: *Proceedings of the Sixth ACM Conference on*

Data and Application Security and Privacy. New Orleans Louisiana USA : ACM, März 2016, S. 26–37. – ISBN 978-1-4503-3935-3

[76] Sun, Chang: *DP-CGANS - Programmcode*. https://github.com/sunchang0124/dp_cgans. Juli 2022. – Zugriffsdatum: 2023-11-07

[77] Sun, Chang ; van Soest, Johan ; Dumontier, Michel: Generating Synthetic Personal Health Data Using Conditional Generative Adversarial Networks Combining with Differential Privacy. In: *Journal of Biomedical Informatics* 143 (2023), Juli, S. 104404. – ISSN 1532-0464

[78] Sweeney, Latanya: K-Anonymity: A Model for Protecting Privacy. In: *International journal of uncertainty, fuzziness and knowledge-based systems* 10 (2002), Nr. 05, S. 557–570

[79] Team-TUD: *CTAB-GAN-Plus-DP - Programmcode*. https://github.com/Team-TUD/CTAB-GAN-Plus-DP. Oktober 2023. – Zugriffsdatum: 2023-11-07

[80] Wang, Xintao ; Yu, Ke ; Wu, Shixiang ; Gu, Jinjin ; Liu, Yihao ; Dong, Chao ; Qiao, Yu ; Change Loy, Chen: Esrgan: Enhanced Super-Resolution Generative Adversarial Networks. In: *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, 2018

[81] Wang, Yaxing ; Zhang, Lichao ; van de Weijer, Joost: Ensembles of Generative Adversarial Networks. In: *arXiv preprint arXiv:1612.00991* (2016), Dezember

[82] Wang, Yu-Xiang ; Balle, Borja ; Kasiviswanathan, Shiva P.: Subsampled Rényi Differential Privacy and Analytical Moments Accountant. In: *The 22nd International Conference on Artificial Intelligence and Statistics*, PMLR, 2019, S. 1226–1235

[83] Wennker, Phil: Machine Learning. In: Wennker, Phil (Hrsg.): *Künstliche Intelligenz in der Praxis: Anwendung in Unternehmen und Branchen: KI wettbewerbs- und zukunftsorientiert einsetzen*. Wiesbaden : Springer Fachmedien, 2020, S. 9–37. – ISBN 978-3-658-30480-5

[84] Wolterink, Jelmer M. ; Dinkla, Anna M. ; Savenije, Mark H. ; Seevinck, Peter R. ; van den Berg, Cornelis A. ; Išgum, Ivana: Deep MR to CT Synthesis Using Unpaired Data. In: *Simulation and Synthesis in Medical Imaging: Second International Workshop, SASHIMI 2017, Held in Conjunction with MICCAI 2017, Québec City, QC, Canada, September 10, 2017, Proceedings 2*, Springer, 2017, S. 14–23

[85] Xie, Liyang ; Lin, Kaixiang ; Wang, Shu ; Wang, Fei ; Zhou, Jiayu: Differentially Private Generative Adversarial Network. In: *arXiv preprint arXiv:1802.06739* (2018)

[86] Xu, Depeng ; Yuan, Shuhan ; Zhang, Lu ; Wu, Xintao: Fairgan: Fairness-aware Generative Adversarial Networks. In: *2018 IEEE International Conference on Big Data (Big Data)*, IEEE, 2018, S. 570–575

[87] Xu, Lei ; Skoularidou, Maria ; Cuesta-Infante, Alfredo ; Veeramachaneni, Kalyan: Modeling Tabular Data Using Conditional Gan. In: *Advances in Neural Information Processing Systems* 32 (2019)

[88] Yang, Yin ; Zhang, Zhenjie ; Miklau, Gerome ; Winslett, Marianne ; Xiao, Xiaokui: Differential Privacy in Data Publication and Analysis. In: *Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data*. Scottsdale Arizona USA : ACM, Mai 2012, S. 601–606. – ISBN 978-1-4503-1247-9

[89] Yoon, Jinsung ; Drumright, Lydia N. ; van der Schaar, Mihaela: Anonymization Through Data Synthesis Using Generative Adversarial Networks (ADS-GAN). In: *IEEE Journal of Biomedical and Health Informatics* 24 (2020), August, Nr. 8, S. 2378–2388. – ISSN 2168-2208

[90] Yovine, Sergio ; Mayr, Franz ; Sosa, Sebastián ; Visca, Ramiro: An Assessment of the Application of Private Aggregation of Ensemble Models to Sensible Data. In: *Machine Learning and Knowledge Extraction* 3 (2021), Dezember, Nr. 4, S. 788–801. – ISSN 2504-4990

[91] Yu, Lantao ; Zhang, Weinan ; Wang, Jun ; Yu, Yong: Seqgan: Sequence Generative Adversarial Nets with Policy Gradient. In: *Proceedings of the AAAI Conference on Artificial Intelligence* Bd. 31, 2017

[92] Zhang, Jun ; Cormode, Graham ; Procopiuc, Cecilia M. ; Srivastava, Divesh ; Xiao, Xiaokui: PrivBayes: Private Data Release via Bayesian Networks. In: *ACM Transactions on Database Systems* 42 (2017), Dezember, Nr. 4, S. 1–41. – ISSN 0362-5915, 1557-4644

[93] Zhang, Xinyang ; Ji, Shouling ; Wang, Ting: Differentially Private Releasing via Deep Generative Model (Technical Report). In: *arXiv preprint arXiv:1801.01594* (2018)

[94] Zhang, Yizhe ; Gan, Zhe ; Carin, Lawrence: Generating Text via Adversarial Training. In: *NIPS Workshop on Adversarial Training* Bd. 21, 2016, S. 21–32

[95] Zhao, Junbo ; Mathieu, Michael ; LeCun, Yann: Energy-Based Generative Adversarial Network. In: *arXiv preprint arXiv:1609.03126* (2016)

[96] Zhao, Zilong ; Kunar, Aditya ; Birke, Robert ; Chen, Lydia Y.: Ctab-Gan: Effective Table Data Synthesizing. In: *Asian Conference on Machine Learning*, PMLR, 2021, S. 97–112

[97] Zhao, Zilong ; Kunar, Aditya ; Birke, Robert ; Chen, Lydia Y.: CTAB-GAN+: Enhancing Tabular Data Synthesis. In: *arXiv preprint arXiv:2204.00401* (2022), April

[98] Zheng, Zhedong ; Zheng, Liang ; Yang, Yi: Unlabeled Samples Generated by Gan Improve the Person Re-Identification Baseline in Vitro. In: *Proceedings of the IEEE International Conference on Computer Vision*, 2017, S. 3754–3762

A Anhang

A.1 Jupyter Notebooks

1. Energie Datensatz
2. Evaluation: Anonymeter

A.2 Quellcode

1. Modell: CTAB-GAN+
2. Evaluation: SDMetrics

A.3 AGMA Spaltennamen

Folgende Auflistung gruppiert die verwendeten Spalten der AGMA Daten anhand ihrer Inhalte:

1. **Angaben zur Person** (11 Spalten)
   - Geschlecht, Alter, Staatsangehörigkeit, Familienstand
   - Bildung: Schulart , Höchster allgemeiner Schulabschluss
   - Beruf: Berufstätigkeit, Beruf, Nettoeinkommen, Arbeitsort, Arbeitsweg
2. **Angaben zum Haushalt** (10 Spalten)
   - Wohnungsart, Haushaltsnettoeinkommen, Anzahl PKWs
   - Personenanzahl: 1-Personen-Haushalt, Personen im Haushalt, Anzahl Kinder
   - Ort: Bundesland, Bundeslandgruppe, Gemeindegrößenklasse, BIK-Regionstyp
3. **Häufigkeit an Einkäufen** (8 Spalten)

- Drogeriemarkt, Getränkemarkt, Baumarkt, Elektrofachmarkt, Discounter, Supermarkt, Großmarkt, Shopping Center

4. **Häufigkeit an Freizeitaktivitäten** (15 Spalten)

   - Medien: Internetnutzung, Fernsehen, Radio, Zeitung, Zeitschrift
   - Unterwegs: Kino, {Theater, Konzert, kulturelle Veranstaltungen}, {Restaurant, Gaststätte, Kneipe, Disco, Club}
   - Kreativität: {Basteln, Heimwerken}, {Stricken, Häkeln, Schneidern}
   - Reise: Letzte größere Ferienreise, Kurzurlaub in den letzten 12 Monaten
   - Verhaltensweisen: Sport treiben, Rauchen, Bier trinken

5. **Transportmittel** (11 Spalten)

   - Häufigkeit der Nutzung: Auto (auch Mitfahrer), Fahrrad, Bahn auf längeren Strecken, Bus bzw. Straßenbahn in der Region, {U-Bahn, S-Bahn oder Regionalbahn in der Region}, Berufliche Flugzeugnutzung, Private Flugzeugnutzung
   - Führerscheinbesitz: PKW, Motorrad, Moped/Mofa
   - Überwiegend genutzte Fahrkartenart

6. **Bewertungen zu Aussagen** (6 Spalten)

   - Markenartikel sind qualitativ besser als markenlose Ware
   - Für besondere Qualität gebe ich gern mehr aus
   - Werbung ist eigentlich ganz hilfreich für den Verbraucher
   - Werbung gibt manchmal recht nützliche Hinweise über neue Produkte
   - Bei den täglichen Einkäufen probiere ich gern mal ein neues Produkt aus
   - Ich bin immer auf der Suche nach Billigangeboten

7. **Dauer außer Haus - Schätzung (Wegezeit)** (4 Spalten)

   - Allgemein, Montags-Freitags, Samstags, Sonntags

8. **Daten zum Interview** (2 Spalten)

   - Monat, Jahr

A.4 Modellparameter

Tabelle A.1 fasst relevante Parameterwerte der verwendeten Modelle zusammen.

GAN	**DPGAN** [85]	**PATE-GAN** [46]	**CTAB-GAN+** [97]	**DP-CGANS** [77]
Netzwerk Architektur				
Datenvorverarbeitung - Kontinuierliche Daten	Skalierung (Bereich [0,1])	Skalierung (Bereich [0,1])	Mode-Specific Normalization	Mode-Specific Normalization
Datenvorverarbeitung - Diskrete Daten	One-Hot-Kodierung	One-Hot-Kodierung	One-Hot-Kodierung & Training-by-Sampling	One-Hot-Kodierung & Training-by-Sampling
Hidden Layer (G)	1 Layer	1 Layer	2 Layer	2 Layer
Aktivierungsfunktion (G)	ReLU	ReLU	ReLU	ReLU
Hidden Layer (D)	1 Layer	Student = 1 Layer Teacher = 1 Layer	4 Layer	2 Layer
Aktivierungsfunktion (D)	ReLU	ReLU	LeakyReLU	LeakyReLU
Komponente C	*irrelevant*	*irrelevant*	4 Hidden Layer & LeakyReLU	*irrelevant*
Anzahl Teacher	*irrelevant*	10	*irrelevant*	*irrelevant*
Softmax	*irrelevant*	*irrelevant*	0,2	0,2
LeakyReLU	*irrelevant*	*irrelevant*	0,2	0,2
Dropout	*irrelevant*	*irrelevant*	0,5	0,5
Netzwerk Training				
Epochenanzahl (keine Privacy)	*irrelevant*	*irrelevant*	Energie: 500, AGMA: 400	500
Batch-Size m	500	64	500	500 (1000 bei $\epsilon = 50$)
PacGAN (Pac)	*irrelevant*	*irrelevant*	*irrelevant*	10
Optimierunsalgorithmus	RMSprop	Adam	Adam	Adam
Lernrate	$5e^{-5}$	$1e^{-4}$	$2e^{-4}$	$2e^{-4}$
Weight Decay	Default: 0	Default: 0	$1e^{-5}$	$1e^{-6}$
Gradient Penality Factor	*irrelevant*	*irrelevant*	10	10
Differential Privacy				
Privatsphären Budget (ϵ)	3, 10, 50	3, 10, 50	3, 10, 50, ∞	3, 10, 50, ∞
Fehlerwahrscheinlichkeit (δ)	$1e^{-5}$	$1e^{-5}$	$1e^{-5}$	$2e^{-6}$
Accountant	RDP-Accountant	Moment Accountant	RDP-Accountant	RDP-Accountant
Noise	Gaussian Noise	Laplacian Noise	Gaussian Noise	Gaussian Noise
Sigma (Gaussian Noise)	2	*irrelevant*	1,02	1

Tabelle A.1: Parameterwahl der verwendeten Modelle

A.5 Modellperformance

Tabelle A.2 stellt die wichtigsten Eigenschaften zur Performance gegenüber.

Datensatz	**Gesamtdauer**	**Anzahl an Epochen**	**∅ ϵ-Anstieg pro Epoche**	**∅ Zeit pro Epoche**
ctab_agma_3	4min	10	0,141	31s
ctab_agma_10	57min	113	0,073	31s
ctab_agma_50	12h und 47min	1.526	0,032	31s
ctab_agma_notPrivate	1h und 20min	400	-	12s
ctab_energie_3	4min	10	0,141	27s
ctab_energie_10	49min	113	0,073	27s
ctab_energie_50	11h und 9min	1.526	0,032	27s
ctab_energie_notPrivate	1h und 10min	500	-	9s
dpcgans_agma_3	9min	14	0,115	42s
dpcgans_agma_10	11h und 25min	142	0,06	292s
dpcgans_agma_50	117h und 7min	921	0,052	458s
dpcgans_agma_notPrivate	1h und 44min	500	-	13s
dpcgans_energie_3	8min	14	0,115	37s
dpcgans_energie_10	11h und 53min	142	0,06	303s
dpcgans_energie_50	118h und 18min	921	0,052	463s
dpcgans_energie_notPrivate	1h und 2min	500	-	7s
dpgan_agma_3	31min	97	0,027	20s
dpgan_agma_10	4h und 47min	863	0,011	20s
dpgan_agma_50	58h und 59min	10.842	0,005	20s
dpgan_energie_3	32min	97	0,027	20s
dpgan_energie_10	4h und 53min	863	0,011	20s
dpgan_energie_50	60h und 26 Minute	10.842	0,005	20s
pate_agma_3	2h und 18min	270	0,0107	31s
pate_agma_10	20h und 27min	2.407	0,004	31s
pate_energie_3	2h und 11min	270	0,0107	29s
pate_energie_10	19h und 48min	2.407	0,004	30s
pate_energie_50	248h und 15min	30.068	0,0017	30s

Tabelle A.2: Modellperformance

Erklärung zur selbstständigen Bearbeitung

Hiermit versichere ich, dass ich die vorliegende Arbeit ohne fremde Hilfe selbständig verfasst und nur die angegebenen Hilfsmittel benutzt habe. Wörtlich oder dem Sinn nach aus anderen Werken entnommene Stellen sind unter Angabe der Quellen kenntlich gemacht.

Ort	Datum	Unterschrift im Original