Aus dem Institut für Medizinische Informationsverarbeitung, Biometrie und

Epidemiologie der Ludwig–Maximilians–Universität München

Vorstand: Prof. Dr. rer. nat. Ulrich Mansmann

Einsatz und Optimierung einer überwachten Klassifizierungsmethode im Kontext eines Privacy-Preserving-Record-Linkage

Dissertation

zum Erwerb des Doktorgrades der Humanbiologie

an der Medizinischen Fakultät der

Ludwig-Maximilians-Universität zu München

vorgelegt von

Daniel Nasseh

aus

München

2014

Mit Genehmigung der Medizinischen Fakultät

der Universität München

Berichterstatter:	Prof. Dr. Jürgen Stausberg
Mitberichterstatter:	Priv. Doz. Dr. Klaus Adelhard
	Priv. Doz. Dr. Stefan Wirth
Dekan:	Prof. Dr. med. Dr. h.c. M. Reiser FACR FRCR
Tag der mündlichen Prüfung:	26.11.2014

Publikationen im Umfeld dieser Arbeit

- Nasseh D, Stausberg J. *Impact of variations in Anonymous Record Linkage on Weight Distribution and Classification*. Stud Health Technol Inform. *2013;192:922.* [PMID: 23920696]

- Nasseh D, Jutta E, Mansmann U, Tretter W, Stausberg J. *Matching study to registry data: maintaining data privacy in a study on family based colorectal cancer.* Angenommen für MIE, Istanbul, September - 2014.

Inhaltsverzeichnis

1. Einleitung

1.1. Einführung in die Thematik

Das Erzeugen, Sammeln und Weitergeben von Daten in großem Stil ist heute selbstverständlicher Bestandteil unseres alltäglichen Lebens geworden. Man denke nur etwa an die vielen bereits in die Milliarden [1] gehenden Online-Profile auf Facebook oder anderen sozialen Netzwerken, auf denen persönliche Daten freiwillig geteilt und veröffentlicht werden [2].

Im Jahr 2013 erregte jedoch die Affäre um unzulässige, weltweite Datenerüberwachung der National Security Agency (NSA) mit der Projektbezeichnung PRISM [3], bei der Daten mit einer Kapazität von mehreren Zettabytes (10^{21} Bytes), einschließlich persönlicher E-Mails und Chatprotokolle, ohne Wissen und Zustimmung erfasst wurden, weltweites Aufsehen [4]. Der Skandal verdeutlicht die Notwendigkeit sicherer Datenschutzkonzepte um geheim zu haltende Daten vor Fremdzugriffen zu schützen.

Gerade in der Medizin kommt dem Datenschutz eine immens hohe Bedeutung zu, da es sich bei medizinischen Daten um Daten mit sensiblen Inhalt (§ 3 Abs. 9 BDSG) handelt. Als sensible Daten bezeichnet man generell Daten mit Angaben über die rassische und ethnische Herkunft, politische Meinung, religiöse oder philosophische Überzeugung, Gewerkschaftszugehörigkeit, Gesundheit oder Sexualleben.

Patientendaten, die solche sensiblen Informationen beinhalten, dürfen unanonymisiert ohne Einverständnis des Patienten nicht veröffentlicht und nur in Sonderfällen weitergereicht werden [5]. Die Sicherheit der Patientendaten ist in Deutschland nicht nur ethisch sondern auch gesetzlich fundiert. Informationen zu Patientendaten fallen nach Artikel § 203 des Strafgesetzbuches (Verletzung von Privatgeheimnissen) unter die ärztliche Schweigepflicht und unterliegen dem Grundrecht auf informationelle Selbstbestimmung [6].

Es stellt sich nun die Frage, wie es im Zuge von medizinischer Forschung ermöglicht werden kann, auf Patientendaten, die einem Schutzversprechen unterliegen, unter Beachtung desselben zuzugreifen. Kohorten-Studien, wie sie beispielsweise im Zuge des KORA- Projektes oder der deutschen Kohorte stattfinden [7,8], arbeiten direkt mit Probanden, die ihre Daten unter Erklärung ihres Einverständnisses zur Verfügung stellen. Die Daten werden hierzu in

vorbereiteten Studienzentren erfasst. Ein Datenzugriff ist also zu Forschungszwecken grundsätzlich möglich.

Schwieriger ist es jedoch, wenn legitimes Forschungsinteresse an Datensammlungen besteht, deren Inhalte ohne explizite Einverständniserklärung des Patienten aufgenommen wurden. Solche Datensammlungen existieren nur dann, wenn es gesetzliche Grundlagen gibt, die die Erfassung medizinischer Daten für die gegebene Einrichtung erlauben. So beschreibt beispielsweise das Bundeskrebsregisterdatengesetz [9] eine dieser Regelungen. Das Tumorregister München etwa (TRM), erfasst sowohl identifizierende (IDAT) als auch medizinische (MDAT) Daten von erkrankten, spezifische Einschlusskriterien erfüllenden, Patienten in München und Umgebung. Datenlieferanten sind hierbei Arztpraxen und Krankenhäuser.

Medizinische Daten innerhalb solcher, nicht auf Patienteneinwilligung basierender Krankheitsregister dürfen nur anonymisiert ausgehändigt werden. Allerdings reicht eine Abtrennung der IDAT von den MDAT oftmals nicht aus. Ort oder Datumsangaben innerhalb der MDAT, wie beispielsweise das Diagnosedatum, können als Quasi-Identifikatoren [5,10] missbraucht werden und somit eine Identifizierung von Personen anhand ihrer MDAT und Hintergrundsinformationen ermöglichen. Über den Health-Insurance-Portability-And-Accountability-Act (HIPAA), eine amerikanische Maßnahme, die sich unter anderem bemüht nationale Standardisierungsregeln zu medizinischen Sicherheitsaspekten zu präsentieren, wird eine gepflegte Liste von Attributen, die als Quasi-Identifikatoren in Frage kämen, zur Verfügung gestellt [11].

Es existieren methodische Ansätze wie K-Anonymity, L-Diversity als auch T-Closeness, die bis zu einem gewissen Grad uneingeschränkte Anonymität garantieren sollen und genannte Gefährdungen seitens Unbefugter auch bei umfangreichem Hintergrundwissen ausschließen sollen [5,10,12]. In der Praxis sind diese Konzepte allerdings oft nur schwer umsetzbar und beschränken durch Generalisierung, Gruppierung, das Einfügen von „Dummy“-Werten und Datenabänderung den Informationsgehalt der Quasi-Identifikatoren bzw. der medizinischen Daten. Ob und in welchem Ausmaß eine Anonymisierung der Patientendaten abseits der Entfernung der IDAT notwendig ist, muss projektspezifisch entschieden werden.

Eine weitere große Herausforderung zeigt sich, wenn medizinische Daten bereits existieren und mit medizinischen Daten aus anderen Datenquellen zusammengeführt werden sollen um etwa mögliche Zusammenhänge zwischen den Daten zu erkennen. Solche Szenarien treten zum Beispiel dann auf, wenn Studiendaten zusätzlich mit Registerdaten verknüpft werden

sollen. Die grundsätzliche Zusammenführung zweier Datensets wird auch als Data-Matching oder Record-Linkage [13] bezeichnet und detailliert unter *Kapitel 1.3.2* beschrieben. Das Matching, also das Zusammenführen der Daten, erfolgt hierbei für gewöhnlich auf der Basis identifizierender Daten wie Namensattributen, Geburtsdatum, Geschlecht und Adresse. Dieser Vorgang ist im Kontext des Zusammenführens von Patientendaten allerdings nicht trivial und unterliegt komplexen Datenschutzmodellen (siehe *Kapitel 1.2.1*), deren Anforderungen es zu erfüllen gilt. So darf unter anderem das Record-Linkage nicht direkt auf den Klartextattributen der IDAT durchgeführt werden. Diese müssen zuerst einwegverschlüsselt werden – das Matching erfolgt also auf einwegverschlüsselten String-Repräsentationen. Ein solches Record-Linkage bezeichnet man dann als Privacy-Preserving-, Anonymous- oder auch Medical-Record-Linkage [14-20].

Als konkretes Beispiel für die Notwendigkeit eines solchen Record-Linkage-Verfahrens stellte sich dem Verfasser dieser Arbeit eine Studie zu familiärem Darmkrebs in München dar (siehe *Kapitel 1.2.1*) [21]. Während der Mitarbeit an der genannten Studie eröffneten sich im Bereich des Record-Linkage einige wissenschaftlich interessante Fragestellungen. Vor allem bezüglich der Klassifizierung, einem wesentlichen Teilbereich des Record–Linkage-Prozesses, konnte Verbesserungspotential bezüglich des Standes der Wissenschaft identifiziert werden, was zu einer Reihe von weiterführenden Untersuchungen, Analysen und Entwicklungen bezüglich der Klassifizierung im Bereich des Privacy-Preserving-Record-Linkage motivierte.

1.2. Motivation zur Durchführung der vorliegenden Arbeit

1.2.1. Studie zu familiärem Darmkrebs

Medizinischer Hintergrund

Bei Darmkrebs, bzw. dem kolorektalem Karzinom, handelt es sich weltweit um die zweithäufigste Tumorerkrankung bei der Frau und die dritthäufigste Tumorerkrankung beim Mann [22]. Verschiedene Risikofaktoren erhöhen die Wahrscheinlichkeit, an Darmkrebs zu erkranken. Als prominent wären schlechte Essgewohnheiten, mangelnde Bewegung, Rauchen und hohes Alter zu nennen [23]. Abgesehen von Risikofaktoren, die auf Umwelteinflüssen basieren, spielen auch genetische Faktoren eine Rolle. Spezifische Gen-Dispositionen die sich in Krankheiten wie z.B. dem Lynch-Syndrom [24] oder dem Gardner-Syndrom [25] ausprägen, erhöhen das Darmkrebsrisiko immens. Der Darmkrebs, der sich normalerweise erst im hohen Alter manifestiert, trifft hierbei oft auch jüngere Personen. Bei familiärem Darmkrebs handelt

es sich hingegen um einen weiteren Risikofaktor, der unabhängig von bekannten genetischen Dispositionen dazu führt, dass diese Erkrankung in Familien oftmals gehäuft auftritt [26].

Die Sterberate nach einer Zeitspanne von fünf Jahren nach der Diagnose des Darmkrebses liegt bei 30%-37% [27]. Für gewöhnlich umfasst die Behandlung, falls möglich, die chirurgische Entfernung des Tumorgewebes, unterstützende Chemotherapie, selten auch in Kombination mit Bestrahlung [28]. Bei rechtzeitiger Erkennung durch Vorsorgeuntersuchungen lässt sich die Sterberate um bis zu 60% verringern [29]. Die Koloskopie ist hierbei die zuverlässigste Methode, aber auch die Kosten sparendere Prüfung auf okkultes Blut im Stuhl kann Hinweise auf Tumorgewebe liefern [30]. Basierend auf den Fakten ist es ersichtlich, welche Konsequenzen eine mangelnde Vorsorge nach sich ziehen kann.

Zielsetzung und grober Ablauf der Studie

Im Rahmen einer Studie zu familiärem Darmkrebs (DKFS: Darmkrebs-Familienstudie), die als Kooperation zwischen dem Institut für Epidemiologie, Biometrie und medizinische Informationsverarbeitung (IBE) an der LMU in München und dem Tumorregister München (TRM: www.tumorregister-muenchen.de) durchgeführt wird, erfolgte eine eingehende Beschäftigung mit der Thematik des familiärem Darmkrebses [21]. Das methodische Hauptinteresse gilt hierbei dem Identifizieren medizinischer Daten von bereits erkrankten Verwandten der für die Studie rekrutierten, neu erkrankten Indexpatienten. Hierdurch sollen Erkenntnisse und Häufigkeiten bezüglich der Thematik ermittelt und gegebenenfalls Empfehlungen und Anpassungen bezüglich der Vorsorge von Angehörigen formuliert werden. Patientendaten zu Tumorerkrankungen werden routinemäßig von Krebsregistern bzgl. eines definierten Einzugsgebietes erfasst. Das Register, aus dem die Studie Daten bezieht, das TRM, umfasst ein Einzugsgebiet von 4,64 Millionen Einwohnern (Stand: 2011) aus den Regionen München und Umgebung.

Leider lassen sich die Familienbeziehungen innerhalb des TRMs nicht rekonstruieren, da notwendige Daten zur Familienstruktur nicht im Register abgespeichert werden. Es gilt also, die im TRM hinterlegten medizinischen Daten (MDAT) der Angehörigen und Indexpatienten mit den Studiendaten, unter Erhalt der Familienstruktur, über andere Wege in Beziehung zu setzen.

Mittels spezieller Erfassungsbögen (siehe *Abbildung 1*) werden die identifizierenden Daten (IDAT) naher Verwandter der neu erkrankten, an der Studie teilnehmenden Indexpatienten im Einzugsgebiet des TRM erfasst.

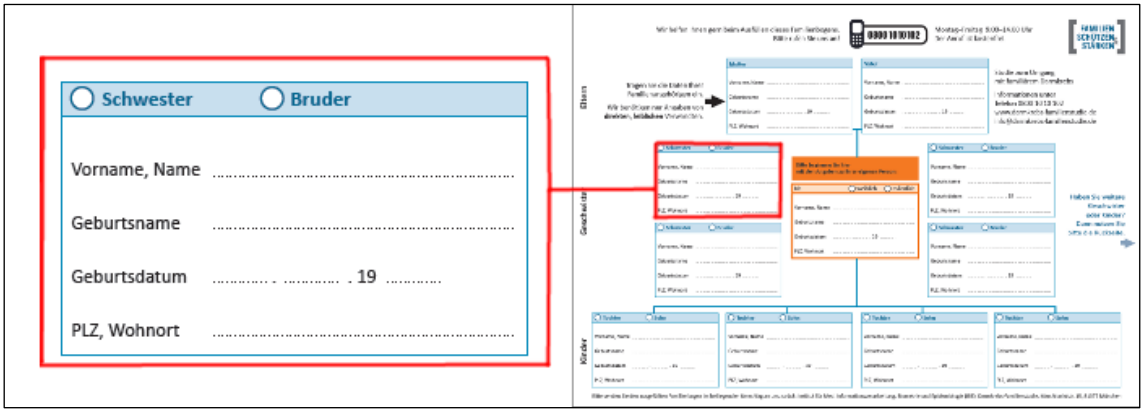


Abbildung 1: Datenerfassungsbogen der DKFS.

Über ein probabilistisches Record-Linkage [31-33] Verfahren (weiterführende Erläuterungen hierzu unter *Kapitel 2.1.2*) lassen sich die hierbei erfassten IDAT der Patienten und Angehörigen zu den im TRM hinterlegten IDAT zuordnen. Die während des Record-Linkage-Prozesses erstellten Links erlauben nachfolgend auch die Zuordnung der MDAT des TRM zu den Studienteilnehmern und ihren Angehörigen. Somit lassen sich Familienstrukturen in den MDAT des TRM rekonstruieren.

Datenschutzkonzept der Studie

Wie unter *Kapitel 1.2* beschrieben, ist nicht nur die Einwegverschlüsselung der Attributwerte Voraussetzung für den sicheren Ablauf eines Privacy-Preserving-Record-Linkage. Studien müssen sich meist nach strengen Datenschutzkonzepten richten. In einer ergänzenden Publikation [34] wurde hierzu ein aus 7 Anforderungen bestehendes Datenschutzmodell vorgestellt, an dem sich die gegebene Studie orientiert. Zentraler Bestandteil dieses Konzeptes ist eine institutionelle sowie organisatorische Trennung der teilnehmenden Parteien in verschiedene Module [35]. Diese Modularisierung resultiert in einer Reihe weiterer Anforderungen und damit verbundener Vorsichtsmaßnahmen, um dem notwendigen Datenschutz zu genügen. *Abbildung 2* beschreibt hierbei vereinfachend den Datenfluss zwischen den wichtigsten an der Studie involvierten Einrichtungen (siehe *Abbildung 2*).

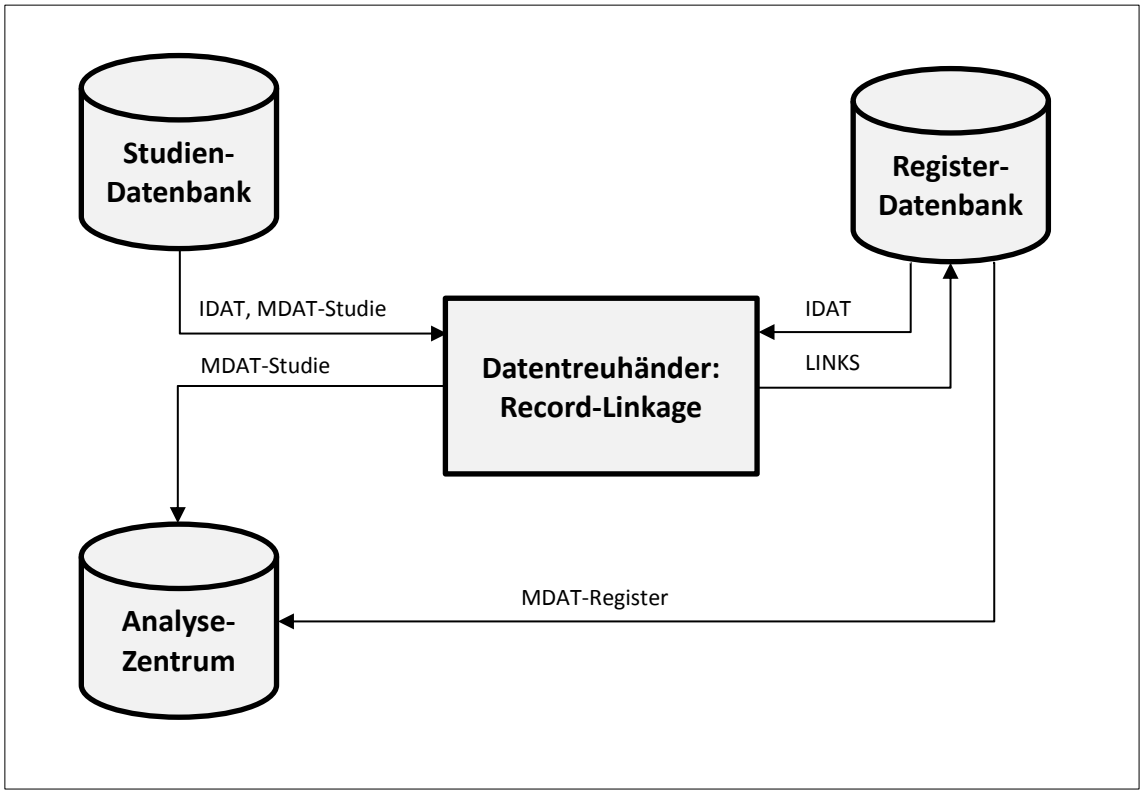


Abbildung 2: Vereinfachtes Datenschutz- sowie Datenflussmodell während der DKFS

Dabei waren abseits der Einwegverschlüsselung der Klartextdaten symmetrische sowie asymmetrische Verschlüsselungsschritte notwendig. Diese wurden konkret mittels AES-Algorithmus mit einer Blocklänge von 128-Bit [36] bzw. AES/RSA mit einer Schlüssellänge von 2048-Bit implementiert [37]. Die technischen Details des Datenschutzmodelles sind für das Verständnis dieser Arbeit allerdings als eher peripher zu verstehen.

1.2.2. Klassifizierungsproblematik während der DKFS

In der DKFS gab es eine Reihe von Aspekten, die im Bezug auf das Privacy-Preserving-Record-Linkage Probleme bereiteten. Ein Hauptproblem zeigte sich bei der Festlegung einer binären Schranke, die die Menge der potentiellen Links in echte bzw. falsche Links unterteilt. Die Festlegung einer binären Schranke ist Teil des Klassifizierungsprozesses des Privacy-Preserving-Record-Linkage, wobei der jetzige Stand der Wissenschaft keine eindeutige, standardisierte Lösung für dieses Problem präsentieren kann [38]. Das Klassifizierungsproblem wird nachfolgend im Bezug auf die Familienstudie eingehend erläutert. *Kapitel 1.3.3* beschäftigt sich zudem mit dem generellen Stand der Wissenschaft zum Klassifizierungsprozess im Bereich des Record-Linkage, insbesondere mit Augenmerk auf binäre Klassifikation (Unterteilung aller Links in zwei Klassen – echte Links und falsche Links).

Während der DKFS wurde primär versucht, manuell eine Klassentrennung zu erreichen. Dies ist eine in der Praxis oft verwendete Methodik [33,39,40]. Grundsätzlich basiert diese auf den Ergebnissen des Matching-Prozesses, also den gesammelten Gewichten der erzeugten Links. Je höher das Gewicht eines Link ist, umso wahrscheinlicher ist es, dass es sich bei den durch IDs repräsentierten Entitäten innerhalb des Link um dieselbe Entität handelt. Allerdings gilt es nun, den Grenzwert zu finden, ab dem ein Link als echter oder falscher Link klassifiziert wird. Die Menge der Gewichte lässt sich wie in *Abbildung 3* illustriert, jeweils als Histogramm darstellen. Dabei gibt die x-Achse die Höhe des Gewichtes an und die y-Achse beschreibt die Häufigkeit eines jeden auftretenden Gewichtes. Um das Histogramm lesbar zu gestalten, sollten die Gewichte gerundet werden – beispielsweise auf die nächste natürliche Zahl. Optimalerweise zeigen sich innerhalb des Histogramms der Gewichte bei guter Datenqualität zwei deutlich voneinander unterscheidbare Erhebungen (*Abbildung 3a*). Nicht nur Genauigkeit und Vollständigkeit definieren in diesem Szenario eine hohe Datenqualität sondern auch Zeitnähe, also ein geringer zeitlicher Abstand bei der Aufnahme der Daten. Diese Erhebungen sind als Klassen zu interpretieren. Die im Histogramm weiter links liegende Erhebung, also diejenige, die niedrigere Gewichte enthält, repräsentiert hierbei falsche Links, die weiter rechts liegende Erhebung echte Links. Ursache für das Auftreten dieser Erhebungen ist, dass Links

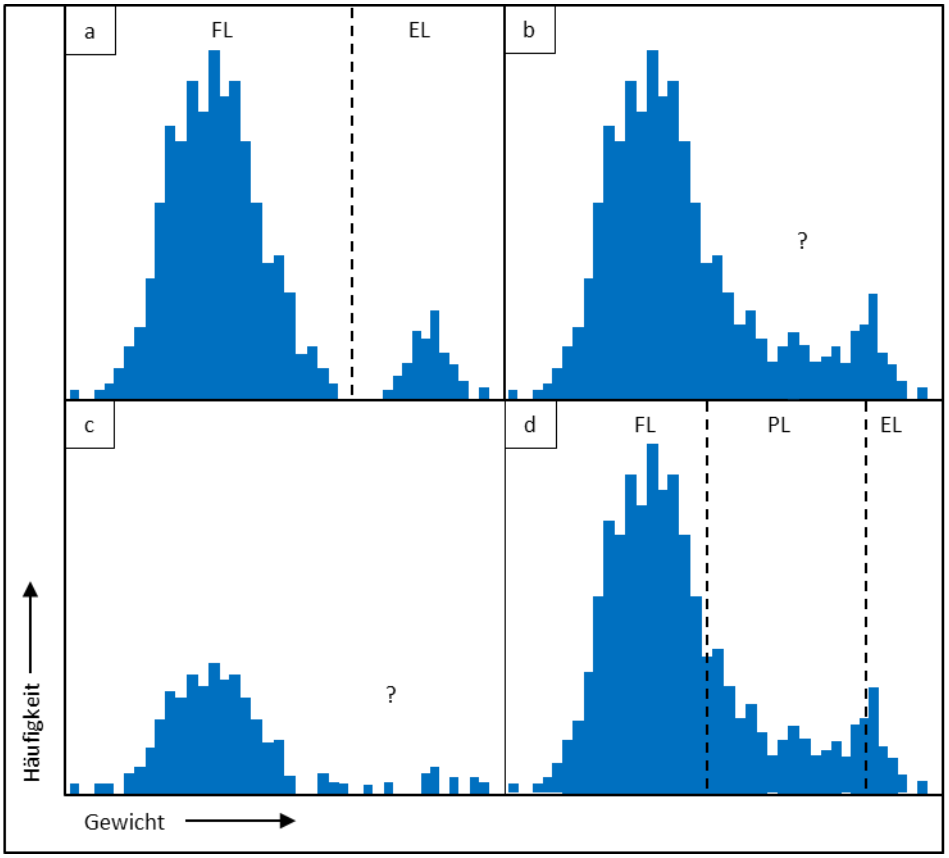


Abbildung 3: Darstellung verschiedener möglicher Histogramme zur Erläuterung der während des Rekord- Linkage auftretenden Klassifikationsproblematik.

innerhalb ihrer Klasse ein ähnliches Gesamtgewicht besitzen, da ähnlich viele Attributswerte übereinstimmen bzw. nicht übereinstimmen. So stimmt bei echten Links meist ein Großteil der Attribute überein, wohingegen bei falschen Links nur wenige oder keine Attribute übereinstimmen. Problematisch in Bezug auf manuelle bzw. unüberwachte Klassifikation [41,42], die sich vollständig an den gegebenen Gewichten orientierten ist im Allgemeinen, wenn es keine klare Klassengrenze gibt oder die Erhebungen nicht markant genug sind, um sie eindeutig voneinander zu unterscheiden (*Abbildung 3b*). Grund hierfür können z.B. mangelnde Datenqualität oder ein hohes Maß an Verwandtschaftsbeziehungen innerhalb der Daten sein. Bei Problemfällen stimmen dann nur einige der Attribute überein, andere wiederum nicht, was in Gesamtgewichten resultiert, die zwischen den Standardwertebereichen für echte bzw. falsche Links liegen. Beispielhaft kann dies anhand von zusammenlebenden Geschwistern dargestellt werden. Deren Daten stimmen im Nachnamen, der Adresse und gegebenenfalls im Geschlecht über, unterscheiden sich jedoch im Vornamen und zumeist im Geburtsdatum (als Ausnahme wären Mehrlinge zu nennen). In solchen Fällen ist es oft schwierig, anhand der Histogramme zu entscheiden, welcher Klasse man diese Links zuordnet. Weiterhin problematisch sind Datensets, zwischen denen nur sehr wenige Übereinstimmungen zu erwarten sind, weswegen anstelle der Erhebung im oberen Gewichtsbereich oftmals durch viele Lücken getrennte Gewichtsanhäufungen zu erkennen sind (*Abbildung 3c*). Hierbei ist es ungewiss, in welche der Lücken ein möglicher Klassentrenner einzutragen wäre.

Im Falle der DKFS war die Klassifikation besonders problembehaftet, da die Daten der Angehörigen der Patienten nicht direkt von den Angehörigen, sondern stellvertretend durch die Indexpatienten über Aufnahmebögen (siehe *Abbildung 1*) oder telefonisch gesammelt wurden. Oftmals fehlten den Patienten hierbei die exakten Informationen, wie beispielsweise der genaue Wohnort, oder das exakte Geburtsdatum ihrer Angehörigen, es wurden jedoch trotzdem Angaben gemacht, die dem nachfolgenden Record-Linkage jedoch eher abträglich waren. Die während des Klassifikationsprozesses erstellten Histogramme während des Record-Linkage zwischen Studien- und Registerdaten entsprachen also nicht dem Optimalbeispiel aus *Abbildung 3a*, sondern eher den Problemfällen wie sie unter *Abbildung 3b* bzw. *Abbildung 3c* wiedergegeben wurden. *Abbildung 4* zeigt diesbezüglich eines der Histogramme der Menge aller Links zum Record-Linkage-Durchlauf am 04.02.2014. Es ist hierbei anzumerken, dass für die Klassifikation innerhalb des DKFS Projektes insgesamt 9 verschiedene Histogramme verwendet werden, die unter anderem eine differenzierte Ansicht von Angehörigen und Patienten erlauben.

Um der Problematik der Unsicherheit zu begegnen, ist es generell, auf datenschutzrechtlich unkritischen Daten, möglich, einen Unsicherheitsbereich explizit zu definieren. Hierzu wird eine weitere Schranke verwendet. Es ist hierbei ausreichend, die beiden Schranken, die den Unsicherheitsbereich aufspannen, grob abzuschätzen (*Abbildung 3d*). Hierbei entstehen drei Klassen. Die der echten Links (oberhalb der oberen Schranke), die der unsicheren/potentiellen Links (zwischen den Schranken), sowie die der falschen Links (unterhalb der unteren Schranke). Die unsicheren Links können dann manuell den echten oder falschen Links zugeordnet werden. Sollte das Vergleichsgewicht zweier echt übereinstimmender Entitäten beispielsweise durch einfache Rechtschreibfehler in den Unsicherheitsbereich gerutscht sein, so lässt sich dies schnell durch die eben genannte manuelle Durchsicht erkennen (*Tabelle 1*). Im dort dargestellten Beispiel würde der Patient mit den Varianten des Nachnamens „SMITH“/“SMYTH“ und kleinem Fehler im Geburtsdatum als identisch identifizierbar sein.

Für solch einen Vergleich sind jedoch Klartextdaten notwendig, welche im Kontext des probabilistischen Privacy-Preserving-Record-Linkage, also auch in Bezug auf die DKFS, nicht gegeben waren. Anhand der hier vorkommenden, einwegverschlüsselten Daten ließ sich lediglich beurteilen, ob Attribute vollkommen übereinstimmen oder nicht. Im Falle der DKFS wurde die Information der einzelnen Attributübereinstimmungen im Unsicherheitsbereich (jedoch ohne Klartextinformation) unterstützend bei der Schrankenfindung mitverwendet (siehe *Abbildung 5*). Die Datei beinhaltete detaillierte Angaben zu linkspezifischen Übereinstimmungen (J), Nicht-Übereinstimmungen(N) und fehlenden Werten auf Seiten der Studiendaten bzw. des TRM (SF=Studie fehlt, TF=TRM Daten fehlen, BF=Daten fehlen auf beiden Seiten).Werte in Klammern standen für die Häufigkeit der jeweils genannten Angaben in Attributen in denen Mehrfachvorkommen möglich sind. Nach Durchsicht der Histogramme wurde die Datei genutzt um die Bestimmung des exakten Punktes des binären Klassifikators zu unterstützen. Im gegebenen Beispiel wurde die Schranke auf 24.9 festgelegt. Der Ausschnitt ist weder in der Zahl der Einträge noch in der Menge der Spalten vollständig.

Tabelle 1: Unterschiedliche Darstellung einer Entität in zwei verschiedenen Datenbanken.

	Datenset 1	**Datenset 2**
Nachname	SMITH	SMYTH
Vorname	ALAN	ALAN
Geburtsdatum	26.02.1983	25.02.1984
Geschlecht	M	M

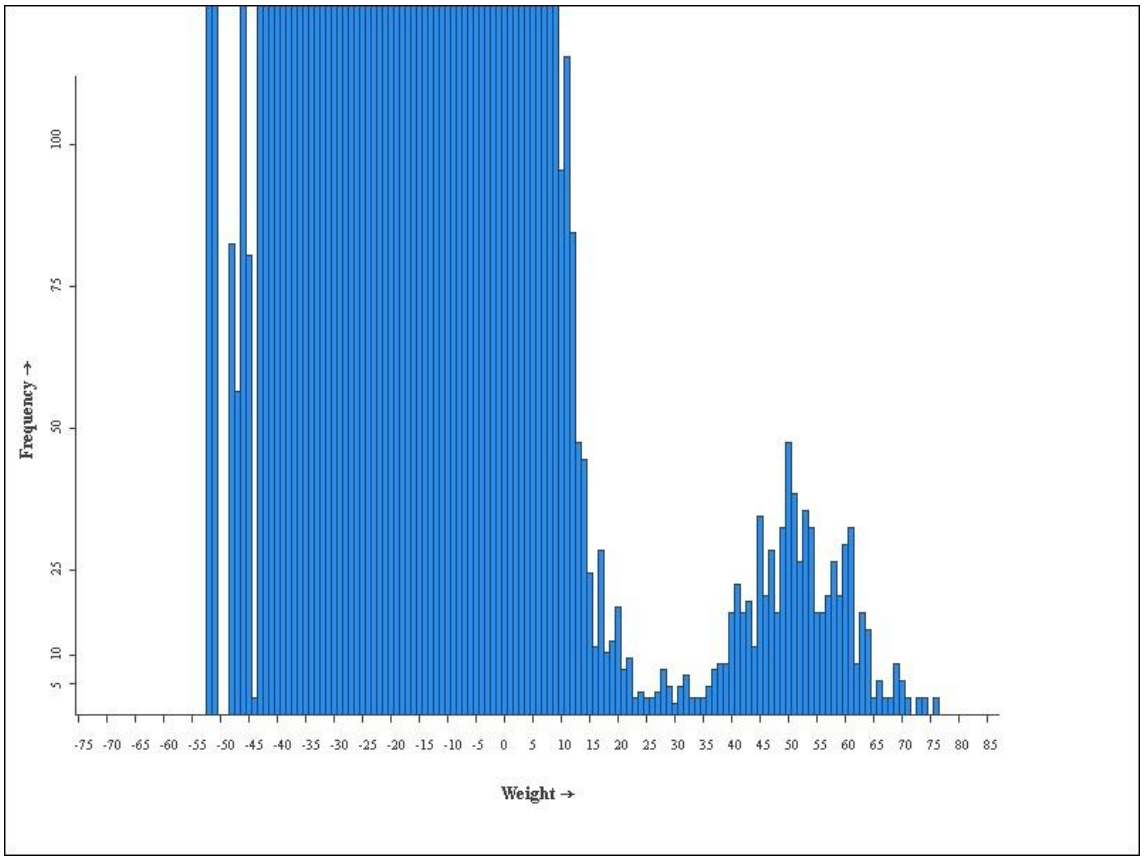


Abbildung 4: Eines der konkreten Histogramme zum Record-Linkage der DKFS am 04.02.2014.

Auch wenn für die DKFS bei der manuellen Schrankensetzung und somit bei einem gewissen Maß an Unsicherheit vorerst verblieben wurde, wäre es wünschenswert, automatisierte, binäre Klassifikationsvarianten entscheidungsunterstützend in den Klassifikationsprozess einzubringen.

Weight	Nachname	Vorname	Geburtsnam	Titel	G-Tag	GMonat	G-Jahr	G-Datum
26,5	[J0,N1,SF0,TF0	J(1)	BF	BF	J	J	J	J
26,3	J(1)	J(1)	BF	BF	J	J	J	J
25	[J0,N1,SF0,TF0	J(1)	[J0,N0,SF0,TF1	BF	J	J	J	J
23,6	[J0,N0,SF1,TF0	J(1)	BF	BF	J	J	J	J
23	[J0,N1,SF0,TF0	J(1)	BF	BF	N	J	J	N
22,7	[J0,N1,SF0,TF0	J(1)	[J0,N0,SF0,TF1	BF	J	J	N	N
22,3	[J0,N1,SF0,TF0	J(1)	BF	BF	J	J	N	N
22,2	J(1)	J(1)	BF	BF	N	J	J	N
20,6	[J0,N1,SF0,TF0	[J0,N1,SF0,TF0	[J0,N0,SF0,TF1	BF	J	J	J	J
20,5	[J0,N1,SF0,TF0	J(1)	BF	BF	J	J	J	J
19,6	J(1)	[J0,N1,SF0,TF0	[J0,N0,SF0,TF1	BF	N	J	J	N
19,6	J(1)	[J0,N1,SF0,TF0	BF	BF	SF	N	N	N
19,2	[J0,N1,SF0,TF0	J(1)	[J0,N0,SF0,TF1	BF	N	J	J	N
18,8	[J0,N1,SF0,TF0	J(1)	[J0,N0,SF1,TF0	BF	J	J	J	J
18,7	[J0,N1,SF0,TF0	J(1)	BF	BF	J	J	N	N
17,7	[J0,N1,SF0,TF0	J(1)	BF	BF	N	N	J	N
17,7	[J0,N1,SF0,TF0	J(1)	BF	BF	J	J	J	J

Abbildung 5: Ausschnitt aus der Pair-Analysis Datei vom Record-Linkage-Durchlauf der DKFS am 19.12.2013.

Leider existieren keine vergleichenden Analysen zu diesen Methoden, und es ist unklar, ob die Methoden überhaupt zur manuellen Klassifikation verbessernd beitragen können. Der Stand der Wissenschaft zu genannten Klassifikationsmethoden wird weiterführend unter *Kapitel 1.3.3* beschrieben.

1.3. Grundlagen des Privacy-Preserving-Record-Linkage

1.3.1. Historischer Hintergrund

Als Record-Linkage bezeichnet man den Prozess des Zusammenführens von Daten verschiedener Datensets. Das Record-Linkage findet dabei in vielen verschiedenen Domänen Anwendung. Das Gesundheitswesen [43,44], nationale Sicherheit [45], Bibliographien (hier auch als Authority-Control [46] bezeichnet) sowie soziale Wissenschaften [47,48] wären hierbei einige der Hauptanwendungsbereiche.

Ein Teilbereich des Record-Linkage, die Klassifikation, spielte in dieser Arbeit die zentrale Rolle. Historisch wurde der Begriff Record-Linkage bereits relativ früh eingeführt. So verwendete Dunn im Jahr 1946 den Begriff zur Beschreibung einer Idee, bei der für jeden Weltenbürger ein Eintrag zu dem als „Book of Life“ bezeichneten Register vorgenommen werden sollte [13]. Im Book of Life sollte jeder Eintrag mit dem Geburtsdatum eines Individuums anfangen und dem Todesdatum enden. Weitere wichtige Eckpunkte des Lebens sollten zwischen diesen zwei Einträgen stehen. Somit gäbe es für jedes Individuum der Erde einen Eintrag im Book of Life, zu dem sich ein Individuum zuordnen ließe- also Grundlage für eine Art universelles Record-Linkage. Zum damaligen Zeitpunkt wäre eine Zuordnung eines Individuums zu diesem Buch relativ schwer gefallen, da es noch keine wissenschaftlich fundierten, automatisierten Methoden gab. Die ersten Ideen hierzu folgten in den 1950ern bzw. frühen 1960ern [49,50], publiziert durch Howard Newcombe. Letzterer ebnete auch den Weg für die ersten probabilistischen Verfahren. Basierend auf seinen Erkenntnissen, dem Berechnen von Gewichten von Übereinstimmungen bzw. Nicht-Übereinstimmungen anhand von Attributshäufigkeiten, formulierten zwei Statistiker, Ivan Fellegi und Alan Sunther, 1969, einen optimalen Algorithmus zum probabilistischen Abgleich von Daten, der auch heute noch weit verbreitet Anwendung findet [31]. So sei zu erwähnen, dass das Record-Linkage-System, das im Methodenteil dieser Arbeit Verwendung fand, auf dem eben genannten Algorithmus beruht. Erwähnenswerte Verbesserungen im Bereich des Record-Linkage konnten noch in den 90er Jahren durch William Winkler erzielt werden [51], der erste Ansätze zur Toleranz von

Variationen in Attributswerten, sowie Möglichkeiten der Abschätzung von Fehlerhäufigkeiten mittels automatisierter Methoden präsentierte.

Das Privacy-Preserving-Record-Linkage basiert auf dem Abgleich von Hash-Werten und entwickelte sich in den 90er Jahren in Frankreich [19,20]. In jüngster Zeit, konkret seit ca. 2010, fandt jedoch eine technische Revolution statt. Im Gegensatz zu den klassischen Methoden, die in diesen Szenarien ihre Vergleiche auf Hash-Werten der zugrunde liegenden Daten ausführten, verwenden die neuen Technologien Bloom-Filter [52] (näher erläutert unter *Kapitel 1.3.2*) als Vergleichsmedium um schließlich Gewichte basierend auf der String-Ähnlichkeit zu approximieren, obgleich die Attributsausprägungen im Klartext nicht lesbar sind. Man spricht hierbei auch von approximativem Record-Linkage. Prominent seien hierzu Arbeiten von Reiner Schnell [14], Elisabeth Durham [53] sowie Peter Christen [54] genannt. Auch wenn diese neuen Technologien vielversprechend klingen und ein definitives, qualitatives Upgrade vor allem in Bezug auf die Sensitivität zum klassischen, probabilistischen Record-Linkage darstellen, gibt es noch viele offene Aspekte, die es hierbei wissenschaftlich zu beleuchten gäbe. Mehrere deutsche Arbeitsgruppen wie beispielsweise das German Record-Linkage-Center (www.record-linkage.de) beschäftigen sich momentan aktiv mit dieser Technologie, und es ist zu erwarten, dass das approximative Record-Linkage bald das probabilistische Record-Linkage im Bereich des Privacy-Preserving-Record-Linkage als den in der medizinischen Forschung verwendeten Standardansatz verdrängt.

1.3.2. Technischer Ablauf des Privacy-Preserving-Record-Linkage

Technisch werden beim Record-Linkage in der Regel Einträge zweier Datensets zueinander zugeordnet. Der Ablauf des Record-Linkage lässt sich in vier rudimentäre Arbeitsschritte einteilen:

- Vorverarbeitung
- Blocking/Indexing
- Gewichtsbestimmung
- Klassifikation

Der eben genannte technische Ablauf wird grafisch in *Abbildung 6* grob wiedergegeben. Die einzelnen Arbeitsschritte werden in den nachfolgenden Unterkapiteln weiterführend beschrieben.

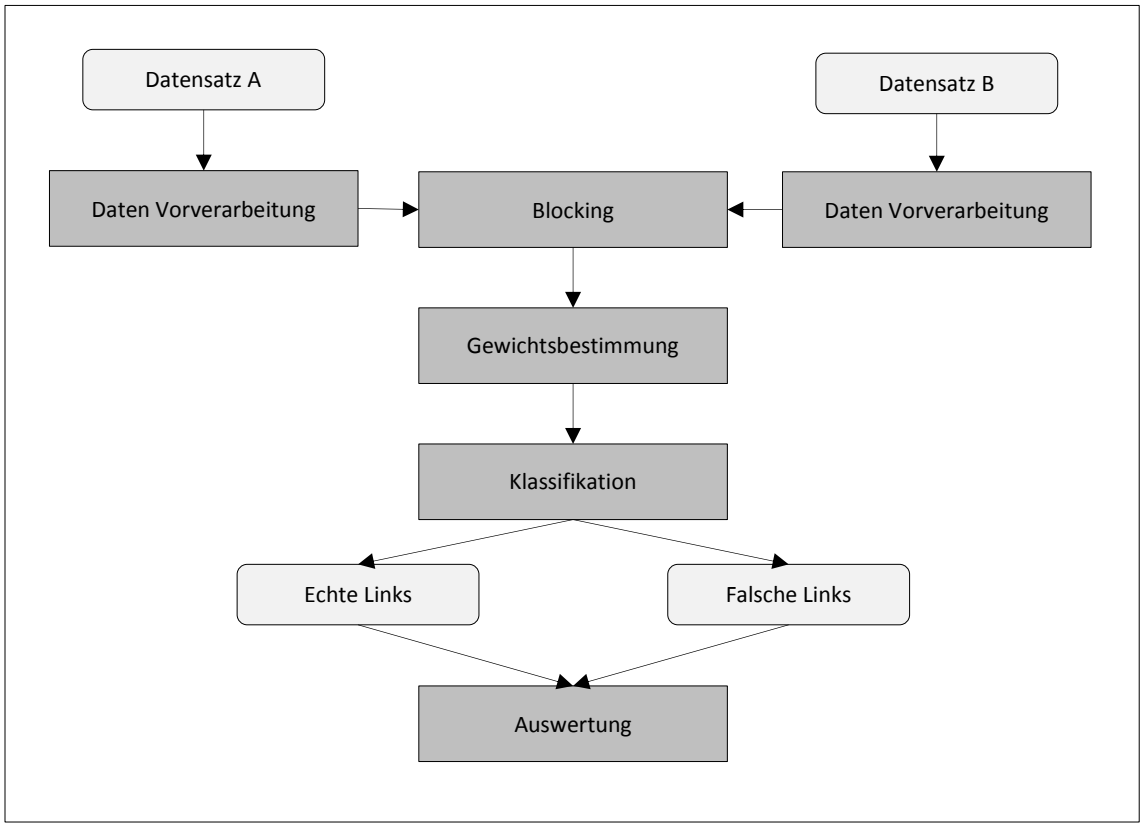


Abbildung 6: Schematischer Ablauf des Privacy-Preserving-Record-Linkage.

Standardisierung (Vorverarbeitung I)

Für gewöhnlich werden Daten vor dem eigentlichen Data-Matching Prozess durch eine Standardisierung der Attributwerte vorverarbeitet. Diese hängt jeweils von der Domäne und Art der Daten ab. So spielt zum Beispiel der Sprachraum, aus dem die Daten stammen, eine entscheidende Rolle. Es gibt also sprachspezifische Varianten zwischen Standardisierungsmethodiken, auch wenn es sich grundsätzlich um dieselbe Art (z.B. Patientendaten) von Daten handelt. Grundsätzlich dient die Standardisierung dazu, Variationen in den verschiedenen Attributswerten gering zu halten und möglichst viele Fehler bereits vor dem eigentlichen Data-Matching auszumerzen.

Bei Personen identifizierenden Daten im medizinischen Sektor werden die ursprünglichen Datenfelder nach bestimmten Regeln standardisiert. Der UNICON-Regelsatz [55] wäre hierbei z.B. der Regelsatz, der in der DKFS Studie inklusive einiger szenarienspezifischer Anpassungen

verwendet wurde. Hierbei sind folgende Anweisungen zu nennen, die während der Standardisierung umgesetzt werden.

- Ersetzung undeutscher Sonderzeichen (basierend auf ausgewählten Listen) in das deutsche Äquivalent (Bsp.: é -> e).
- Entfernung ungeeigneter Zeichen. Dies betrifft Symbole, die im jeweiligen Feld nicht auftreten sollten (Bsp.: Hans-Wagne%r -> Hans-Wagner).
- Uniforme Großschreibung (Bsp.: Hans-Wagner -> HANS-WAGNER).
- Umlaut-Normalisierung (Bsp.: FÖRSTER -> FOERSTER).
- Ersetzung von Trennsymbolen durch Leerzeichen (Bsp.: HANS-WAGNER -> HANS WAGNER).
- Erkennung spezifischer Schlagwörter. Dieser Schritt ist feldspezifisch. Im Feld „Titel“ werden hierbei beispielsweise nur gültige Titel (basierend auf einer zuvor erstellten Liste) zur weiteren Verarbeitung zugelassen. (Bsp.: Dr.)
- Konsistenz- bzw. Formatprüfung. (Bsp.: Entfernung des Geburtsdatums bei 33.02.19083)
- Bei Attributen mit möglicher Mehrfachausprägung (z.B. Doppelname): Aufteilen der Felder in neue Attributgruppen. (Bsp.: HANS WAGNER -> VORNAME 1: HANS/VORNAME 2: WAGNER).

Weiterhin ist es möglich, nach phonetischen Kriterien zu standardisieren. Somit werden Namensvarianten wie beispielsweise „Meyer“, bzw. „Meier“, die phonetisch übereinstimmen, in eine standardisierte Variante umgewandelt. Algorithmen, die hierzu verwendet werden, sind im englischsprachigen Raum der SOUNDEX [56] bzw. im deutschsprachigen Raum die Kölner Phonetik [57].

Einwegverschlüsselung (Vorverarbeitung II)

Ein weiterer Schritt der Vorverarbeitung fällt ausschließlich beim Privacy-Preserving-Record-Linkage an. Es handelt sich hierbei um die Einwegverschlüsselung der Daten, die basierend auf ausgewählten Algorithmen einwegverschlüsselt werden müssen bevor sie abgeglichen werden dürfen. Beim deterministischen, bzw. dem probabilistischem Record-Linkage werden zu jedem standardisiertem Attributswert anhand von Hash-Funktionen mathematisch nicht umkehrbare Bit-Sequenzen, die sich beispielsweise als Hexadezimalcode darstellen lassen, erzeugt. Man spricht hierbei von Kontrollnummern [58-60]. Als Besonderheit sei zu nennen, dass moderne Hash-Funktionen in der Regel, ausgehend vom Ausgangswert, nahezu immer verschiedene Hash-Werte erzeugen. Zu jedem Ausgangswort gibt es also meist exakt einen spezifischen

Hash-Wert. Sollte es dennoch Hash-Werte geben, die zu verschiedenen Eingabewerten passen, spricht man von Kollisionen [61], die aber extrem selten vorkommen. Zu älteren Hash-Funktionen wie dem MD5 wurden bereits Kollisionsfunde gemeldet. Diese gelten somit als veraltet und sollten nicht weiter verwendet werden, wohingegen Algorithmen aus der SHA-2 oder noch besser aus der SHA-3 Familie dem aktuellen Sicherheitsstand entsprechen [62,63].

Tabelle 2 illustriert die Ausgabe zu verschiedenen Eingabewerten in Hexadezimalschreibweise, basierend auf der SHA-256 Funktion. Trotz der hohen Textähnlichkeit der Ausgangswerte im vorliegenden Beispiel erzeugt die Hash-Funktion komplett unterschiedliche Rückgabewerte.

Tabelle 2: Anwendung des SHA-256 auf verschiedene Ausgangswerte.

Ausgangswert	**Hash-Wert**
Meier	05c2d2b4cad1a3f5bf547b484ac6f4a70893e944d5bd6fe0f28db40453bf3f3c
Meyer	876fdfa1d1152c1d024386a1f66e7725f292ef83404fc4d3be79c1b51cc81c45

Auf den Hash-Werten ist zwar immer noch ein Abgleich möglich, allerdings sind die Daten nur noch über einen Wörterbuchangriff identifizierbar und in den ursprünglichen Klartext rücküberführbar. Bei einem Wörterbuchangriff werden Wertelisten mit derselben Hash-Funktion des unter Angriff stehenden Datensatzes einwegverschlüsselt. Dies ermöglicht ein Mapping der Hash-Werte dieser Werteliste und des unter Angriff stehenden Datensatzes. Konsequenterweise sollte der exakte Hash-Algorithmus nicht bekannt gegeben werden, oder es sollten spezielle Schlüssel verwendet werden, die die Ausgangsfunktion modifizieren. Man spricht hierbei auch von Hash-based Message Authentication Code Verfahren (HMAC) [64]. Alternativ lässt sich auch nach geheim gehaltenen Regeln sogenanntes „Salz“, einfache Buchstaben oder Zahlenketten, an die Ausgangswerte anhängen, was einen weiteren Schutz gegenüber Wörterbuchangriffen darstellt [65].

Das approximative Record-Linkage, das eine Weiterentwicklung des probabilistischen Privacy-Preserving-Record-Linkage darstellt, ersetzt die Einwegverschlüsselung basierend auf Hash-Werten durch Bloom-Filter [14,52]. Bloom-Filter sind Bit-Arrays, also Speicherstrukturen mit einer festgelegten Länge und einer Indexstruktur. Die Feldwerte des Arrays lassen sich dabei mit Bit-Werten, also mit 0 oder 1, belegen.

Initialisiert werden die Bloom-Filter in jedem Feld mit einem 0-Wert. Die Technik basiert darauf, die zu verschlüsselnden Wortketten in Q-gramme (in der Regel Bi-gramme) zu zerlegen.

Auf jedes Q-Gramm werden dabei mehrere Hash-Funktionen angewandt. Nach Kirsch et. Al [66] sind zwei Hash-Funktionen ausreichend. Der Rückgabewert dieser Hash-Funktionen muss ein Wert zwischen 0 und der Länge des Bloom-Filter sein. Diese Rückgabewerte geben nun den Index wieder, an dem der Bloom-Filter mit einer 1 belegt werden soll. Eine erläuternde graphische Darstellung findet sich hierzu in *Abbildung 7*. In diesem Beispiel werden die Namensausprägungen „Anna“ und „Anne“ in Bi-Gramme zerlegt auf die jeweils eine Hashfunktionen angewendet wird. Die Hashfunktion gibt jeweils einen Rückgabewert an der den Index spezifiziert an dem der jeweils vorliegende Bloom-Filter mit dem Bit-Wert 1 belegt wird.

Die Berechnung der Gewichte sowohl beim Kontrollnummer- als auch auf Bloom-Filter-Abgleich wird im nachfolgenden Unterkapitel zur Gewichtsberechnung weiter diskutiert.

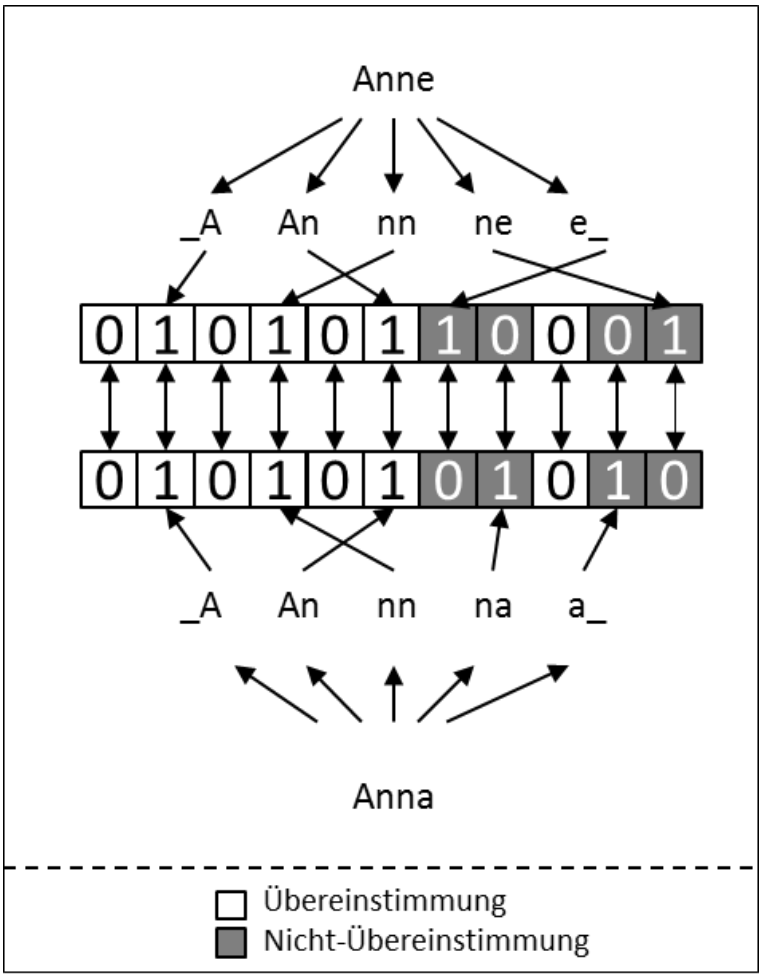


Abbildung 7: Einwegverschlüsselung von Werteausprägungen anhand von Bloom-Filtern.

Blocking/Indexing

Die Zuordnung von Einträgen innerhalb zweier Datensets A und B erfolgt im Grunde genommen durch den Abgleich jeweils eines Eintrages aus A mit allen Einträgen aus B. Die Menge an notwendigen Vergleichen ist also das Kreuzprodukt der Anzahl an Einträgen aus A und B:

$$|A| \times |B|$$

Würde man beispielsweise die Einwohner zweier größerer Städte (1 Mio. Einwohner) miteinander abgleichen wollen, würde dies in einer Billionen (10^{12}) individuellen Vergleichen und Gewichtsberechnungen resultieren. Geht man also wie gegeben vor, kann der rechenintensive Aufwand oftmals das Limit der gegebenen Hardware bzw. gegebene Zeitlimits übersteigen. Abhilfe hierzu schafft die Verwendung von sogenannten Indexing/Blocking-Techniken. Am prominentesten wäre die Verwendung von Blocking-Variablen zu nennen. Zwar werden auch hier auf dem Kreuzprodukt der Einträge Vergleiche ausgeführt, Gewichte werden jedoch nachfolgend nur berechnet, wenn die verglichenen Einträge in zumindest einer der gegebenen Blocking-Variablen übereinstimmen. Es ist abzuraten, sich lediglich auf eine Blocking-Variable zu beschränken. Hierbei kann es passieren, dass Rechtschreibfehler oder andere Abwandlungen in Attributwerten einer in zwei Datensätzen repräsentierten Entität (wie z.B. Änderung des Nachnamens bei Hochzeit) dazu führen, dass diese nicht bei der Gewichtsberechnung berücksichtigt werden. In der Praxis verwendet man deswegen mehrere Blocking-Variablen [19], wie beispielsweise, den phonetischen Nachnamen sowie das Geburtsdatum. In der DKFS zu familiärem Darmkrebs wurden als Blocking-Variable der phonetische Nachname, der phonetische Vorname sowie das Geburtsjahr ausgewählt.

Die meist angewandte Variante des Blockings beschreibt das Standard-Blocking [31], bei der die Blocking-Variablen der Vergleiche genau übereinstimmen müssen, damit ein Gewicht weiterführend berechnet wird. Hierbei ergeben sich Varianten. Stimmen zwei Einträge in mehreren Blocking-Variablen überein, kann dasselbe Gewicht für einen Vergleich mehrfach berechnet werden. Verwendet man also einfache Listenstrukturen und hängt dort die Informationen zu Links und ihren Gewichten aneinander, so können Einträge mehrfach, entsprechend der Anzahl der Blocking-Variablen auftreten. Verwendet man Hash-Strukturen, die IDs der Links als eindeutigen Schlüssel verwenden, werden die Gewichte nur einfach abgespeichert. Dieses Phänomen und die Auswirkungen auf die nachfolgende Klassifikation wurden vom Autor in einer dieser Arbeit vorhergehenden Publikation näher untersucht [67].

Weitere Varianten, die den Rechenaufwand des Blockings einschränken, ergeben sich aus dem Sortieren der Datensätze. Hierbei wäre der Sorted-Neighbourhood-Approach zu nennen bei dem mittels eines Sliding-Windows mit fester Größe über die alphabetisch sortierte Datenbank gefahren wird und Teilwortketten die innerhalb des Sliding-Windows übereinstimmen zum Blockingabgleich verwendet werden. [68,69]

Beim Canopy-Clustering [70] werden Werte, die sich in der Blocking-Variable ähneln, in denselben Cluster eingefügt und innerhalb dieses Clusters abgeglichen. Dieses Verfahren ist allerdings nicht auf einwegverschlüsselte Daten übertragbar, da die verwendeten Ähnlichkeitsmaße Klartextdaten voraussetzen.

Als Nebeneffekt hat das Blocking auch Einfluss auf Qualitätswerte, vor allem auf die Anzahl der True-Negatives, die zur Evaluation des Record-Linkage verwendet werden können (siehe *Kapitel 1.3.4*). Da bei Anwendung von gut gewählten Blocking-Variablen die Anzahl der True-Positives, False-Positives sowie False-Negatives meist nur leicht variiert, sich aber in der Anzahl der True-Negatives gewaltig reduziert, ist vor allem die Spezifität hiervon betroffen. Da die Spezifität beim Record-Linkage meist jedoch nahe der 100% liegt, verwendet man aber generell lieber den F-Measure-Wert, der unabhängig von der Spezifität, bzw. von den True-Negatives fungiert [71].

Gewichtsbestimmung

Während des Blockings werden Eintragsvergleiche ausgewählt, zu denen es zu bestimmen gilt, ob diese Vergleiche tatsächlich übereinstimmen oder nicht. Hierfür werden beim Privacy-Preserving-Record-Linkage die individuellen Kontrollnummern bzw. Bloom-Filter der Einträge verglichen. Insgesamt gibt es hierbei drei verschiedene Herangehensweisen. Die triviale Variante stellt das deterministische Record-Linkage dar. Hierbei werden zwei Einträge jeweils als echter Link klassifiziert, falls alle Kontrollnummern paarweise exakt übereinstimmen. Im Gegensatz zu den anderen Varianten entfällt also beim deterministischen Record-Linkage eine weiterführende Klassifizierung, eine Gewichtsbestimmung im eigentlichen Sinne findet nicht statt. Die Methodik erzielt in der Regel Spezifitätswerte von 100%, allerdings werden sämtliche echte Links, die nur geringfügig voneinander abweichen, übersehen. Zwar kann gute Standardisierung diese Fehler teilweise beseitigen, grundsätzlich liefert die Methodik jedoch Ergebnisse mit einer vergleichsweise mangelhaften Sensitivität [53]. Ein prominentes Beispiel für die Implementierung eines deterministischen Record-Linkage Systems ist der PID-Generator der Technologie- und Methodenplattform für die vernetzte medizinische Forschung (TMF) [72], der grundsätzlich jedoch eher als Pseudonymisierungs-Instrument zu verstehen ist.

Im Gegensatz zum deterministischen Record-Linkage stellt sich das probabilistische Record-Linkage als fehlertoleranter dar. Hierbei wird für jeden paarweisen Abgleich der Kontrollnummern zwischen den zu vergleichenden Einträgen ein Einzelgewicht berechnet und anhand der Summe dieser Einzelgewichte wird der Eintrag als echter bzw. falscher Link klassifiziert (siehe *Formel 1*).

$$w = \sum w_i \qquad (1)$$

Für die Erläuterung der Berechnung der Einzelgewichte sind einige initiale Definitionen notwendig. Während A und B die zu vergleichenden Datensets repräsentieren, stehen die Mengen M und U für die Menge der Übereinstimmungen bzw. der Nicht-Übereinstimmungen (siehe *Formel 2-4*).

$$A \times B = \{(a, b); a \in A, b \in B\} \qquad (2)$$

$$M = \{(a, b); a = b, a \in A, b \in B\} \qquad (3)$$

$$U = \{(a, b); a \neq b, a \in A, b \in B\} \qquad (4)$$

Bei $a_1,\ldots,a_n$ bzw. $b_1,\ldots,b_n$ handelt es sich um die einzelnen Attribute zu den Einträgen a bzw. b, aus Datenset A bzw. B (siehe *Formel 5*).

$$a = (a_1, \ldots, a_n), b = (b_1 \ldots, b_n) \qquad (5)$$

Nach Fellegi und Sunther resultieren Übereinstimmungen in den Ausprägungen in einem positiven Einzelgewicht, Nicht-Übereinstimmung in einem negativen Einzelgewicht [31]. Die Höhe des Gewichts wird von der Häufigkeit der zu vergleichenden Werteausprägung bzw. der abgeschätzten Fehlerhäufigkeit in diesem Attribut beeinflusst. Die Häufigkeit der Kontrollummern-Ausprägungen wird dabei formell durch die sog. u-Werte repräsentiert (siehe *Formel 6*) [31,39].

$$u_{ik} = P(a_i = b_i \wedge a_i = x_{ik} | (a, b) \in U) \qquad (6)$$

Der u-Wert beschreibt konkret die Wahrscheinlichkeit, dass zwei Einträge im Merkmal i mit der Ausprägung x_{ik} übereinstimmen und es sich dabei nicht um dieselbe Person/Eintrag handelt. Die u-Werte lassen sich hierbei im praktischen Umgang direkt aus der Häufigkeit von

zugrunde liegenden Populationen bzw. direkt aus den Datensets ableiten [39]. Kommt beispielsweise der Vorname „Peter“ im zugrunde liegenden Datenset mit Größe 10.000 insgesamt dreimal vor, dann beträgt der u-Wert der Ausprägung „Peter“ 3/10.000). Da die Datensätze im Kontext des Record-Linkage in der Regel aus jeweils zwei Daten-Quellen bestehen können die Datenquellen hierfür vereinfachend vereint werden.

Die in den Ausprägungen auftretenden Fehlerhäufigkeiten, die ebenfalls zur Gewichtsberechnung benötigt werden, lassen sich durch die m-Werte repräsentieren (siehe *Formel 7*).

$$m_{ik} = P(a_i = b_i \wedge a_i = x_{ik} | (a, b) \in M) \qquad (7)$$

Der m-Wert beschreibt hierbei konkret die Wahrscheinlichkeit, dass zwei Einträge im Merkmal i mit der Ausprägung x_{ik} übereinstimmen und es sich dabei um die selbe Person/Eintrag handelt. Die m-Werte lassen sich vereinfacht jedoch auch als invertierte Fehlerhäufigkeiten im jeweiligen Attribut interpretieren. Typischerweise haben Adressangaben eine relativ hohe Fehlerhäufigkeit. Würde man also zum Beispiel in einem Datensatz zu 10% der Fälle Fehler in den Adressangaben erwarten, wäre der hierzu gehörende m-Wert 0.9. Die m-Werte können entweder aus ähnlichen [39,73], bereits ausgewerteten Datenbeständen mit bekannten Fehlerhäufigkeiten oder mittels einer Variante des Expectation-Maximation-Algorithmus [74] abgeschätzt werden.

Anhand der u-Werte und m-Werte lassen sich schließlich die Einzelgewichte berechnen (siehe *Formel 8* bzw. *Formel 9*).

$$w_i = \log\left(\frac{m_i}{u_{ik}}\right), falls\ a_i = b_i \wedge a_i = x_{ik} \qquad (8)$$

$$w_i = \log\left(\frac{1 - m_i}{1 - u_{ik}}\right), falls\ a_i \neq b_i \wedge a_i = x_{ik} \qquad (9)$$

Falls die vergleichenden Attributsausprägungen übereinstimmen, wird wie bereits erwähnt ein positives Gewicht berechnet, falls die vergleichenden Attributsausprägungen nicht übereinstimmen, wird ein negatives Gewicht berechnet. Zudem gilt: Stimmen Kontrollnummern in einer seltenen Ausprägungen überein, so resultiert dies in einem stärkeren Gewicht. Das Übereinstimmen in häufigen Ausprägungen kann eher auf Zufall basieren, demnach wird ein niedrigeres Gewicht vergeben. Je höher die abgeschätzte Fehlerrate in einem Attribut ist, umso unbedeutender, also niedriger ist das Gewicht im

Vergleich zu anderen Attributen mit geringeren Fehlerraten. Nach Aufaddieren der Einzelgewichte zu einem Gesamtgewicht kann schließlich klassifiziert werden.

Liegt das Gesamtgewicht eines Links über einem spezifischen Schrankenwert, so wird er als echter Link bewertet, unterhalb dieser Schwelle als falscher Link. Man spricht hierbei von einer binären Klassifikation (hierzu mehr unter *Kapitel 1.3.2*.). Das Auffinden dieses Schrankenwertes war eine nicht triviale Aufgabe und Hauptthematik dieser Arbeit. Im Gegensatz zum deterministischen Record-Linkage unterscheidet sich also das probabilistische Record-Linkage darin, dass es nicht in allen Kontrollnummern exakt übereinstimmen muss und somit zu einem gewissen Grad Fehler in den Daten zulässt. Die Spezifität erleidet hierbei in der Regel nur geringfügige Einbußen und liegt je nach Datensatz nahe 100%. Die Sensitivität kann durch das Tolerieren weniger Unstimmigkeiten im Vergleich zum deterministischen Record-Linkage enorm verbessert werden und liegt je nach Datensatz, nach einem systematischem Review von Silveira [75] bei den ausgewerteten Arbeiten zwischen 74-98%.

Dennoch besitzt das probabilistische Record-Linkage auf einwegverschlüsselten Daten Schwächen. Durch die Einwegverschlüsselung ist es grundsätzlich nicht möglich, die Ähnlichkeit zweier Ausprägungen zu gewichten. Da bereits kleine Fehler in den Werteausprägungen (z.B. Schmitt bzw. Schmidt) zu komplett unterschiedlichen Hash-Werten führen, ist es lediglich möglich, zu bewerten, ob die Werte übereinstimmen oder nicht (siehe hierzu auch *Kapitel 1.2.2*).

Das approximative Record-Linkage tritt dieser Problematik entgegen. Wie bereits unter *Kapitel 1.3.2* beschrieben, werden die Ausprägungen alternativ zu den vorhergehenden Methoden mittels Bloom-Filtern einwegverschlüsselt. Der Abgleich erfolgt also nicht mehr wie bei den Vorgängervarianten auf Hash-Werten sondern auf den Bloom-Filtern. Dabei kann nicht nur wie auf Hash-Werten festgestellt werden, ob Werte generell übereinstimmen, sondern auch, wie sehr sich zwei Bloom-Filter ähneln.

Die Distanz zweier Bloom-Filter zueinander lässt sich mittels des Dice-Koeffizienten (siehe *Formel 10*), berechnen, der sich als passendes Distanzmaß bewiesen hat [14,76].

$$D_{A,B} = \frac{2h}{(a+b)} \qquad (10)$$

Auf das Szenario des approximativen Record-Linkage übertragen entsprich h der Anzahl an Bitpositionen, die in beiden zu vergleichenden Bloom-Filtern (A,B) mit 1 belegt wurden, a ist die Anzahl an Bitpositionen, die ausschließlich in A mit 1 belegt wurden, wohingegen b die

Anzahl an mit 1 belegten Bitpositionen in B wiedergibt. Angewandt auf das Beispiel aus *Abbildung 7* ergibt sich ein Dice-Koeffizient von $\frac{6}{10}$. Der Rückgabewert der Distanzfunktion liegt hierbei zwischen 0 und 1, wobei ein hoher Wert für eine hohe Ähnlichkeit steht. Da Feldwerte in den Bloom-Filtern mehrfach belegt werden können, lässt sich die Ähnlichkeit nicht in selben Ausmaß wie bei String-Vergleichen im Klartext bestimmen. Die Übereinstimmung wird demnach approximiert. Daher auch der Name: approximatives Record-Linkage. Die Einzelgewichte werden schließlich, wie im Falle des probabilistischen Record-Linkage, zu einem Gesamtgewicht aufaddiert. Der Klassifikationsprozess verläuft demnach zwischen beiden Methoden analog. Es gibt noch viele offene Fragestellungen, die zu dieser in stetiger Weiterentwicklung befindlichen Technologie Klärung benötigen. So gab es Ende 2013 beispielsweise noch keine publizierten Aussagen darüber, mit welchem Faktor die auf Bloom-Filter-Vergleich beruhenden Einzelgewichte zu verrechnen wären. Beispielsweise sollte der Nachname eine höhere Gewichtung besitzen als die Postleitzahl, da sich diese im Verlauf des Lebens öfters ändern kann. Dies wäre nur eines der Probleme, die im klassischen probabilistischen Record-Linkage bereits gelöst wurden, weswegen das approximative Record-Linkage zu diesem Zeitpunkt noch nicht unangefochten als Standardvariante für Privacy-Preserving-Record-Linkage zu interpretieren wäre. Vergleichende Arbeiten haben jedoch gezeigt, dass das approximative Record-Linkage durch die Beurteilung der Ähnlichkeit das Potential besitzt, die älteren Varianten in Bezug auf die Qualität des Matchings, vor allem was die Sensitivität betrifft, zu überflügeln [18]. Ob und inwiefern Gewichtungen des probabilistischen Record-Linkage auf das approximative Record-Linkage übertragbar sind, ist Aufgabe aktueller Forschung.

Binäre Klassifikation

Im Falle des Privacy-Preserving-Record-Linkage ist durch die Einwegverschlüsselung der Ausgangsdaten oftmals eine manuelle Zuordnung unsicherer Links nicht möglich (*siehe Kapitel 1.2.2*). Dies resultiert in der Notwendigkeit von binärer Klassifikation, also im Normalfall in der Bestimmung eines spezifischen Schrankenwertes, der die Menge der Links, basierend auf ihrem Gewichtswert, in echte bzw. falsche Links einteilt. Die binäre Klassifikation ist jedoch nicht nur im Falle von unter Datenschutz befindlicher Daten notwendig sondern auch beim Einsatz vollautomatischer Systeme, bei denen keine manuelle Nachkontrolle möglich ist.

Die Rückgabe des probabilistischen bzw. approximativen Record-Linkage ist eine Liste von Links, bestehend aus einem Paar von Datenset spezifischen IDs, die eine Referenz auf den im jeweiligen Datenset beinhalteten Eintrag darstellen, sowie ein assoziiertes Gewicht, das

Aufschluss darüber gibt wie gut die beiden referentiellen Einträge zueinander passen (siehe *Tabelle 3*).

Tabelle 3: Beispielhafte Darstellung des Inhaltes einer Gewichtsdatei.

ID A	ID B	Gewicht
1252	5332	76,74
1773	6784	74,33
34	588	71,22
788	899	55,39
1899	1754	23,76

Basiert die Klassifikation ausschließlich auf der Verteilung der genannten Gewichte, spricht man von unüberwachter Klassifikation. Werden von dieser Verteilung unabhängig Trainingsdaten verwendet spricht man typischerweise von überwachter Klassifikation. Zudem existieren auf Regeln basierende Klassifikationsmethoden, die in beiden der vorhergehenden Ansätze unterstützend genutzt werden können, aber auch als eigenständige Methodik existieren.

1.3.3. **Klassifikationstechniken**

Unüberwachte Klassifikation

Die unüberwachten Methoden richten sich vollständig nach der Verteilung und den Häufigkeiten der Gewichte, die sich auch als Histogramm illustrieren lassen. Zur Histogramm-Erstellung werden die Gewichtswerte auf einen spezifischen Wert gerundet (beispielsweise auf natürliche Zahlen) und entsprechend der Häufigkeit dieses Wertes in das Histogramm eingetragen. Bei qualitativ hochwertigen Daten zeigen sich hierbei im Histogramm der Gewichte oftmals zwei Erhebungen, die sich leicht manuell voneinander trennen lassen (siehe *Abbildung 3a*). Unabhängig vom Histogramm, aber basierend auf denselben Daten kann diese Trennung auch durch verschiedene automatisierte Algorithmen, wie z.B. aus dem maschinellen Lernen bekannte Clustering-Verfahren erfolgen [77-78]. Der Erfolg der unüberwachten Klassifizierung hängt demnach stark von der Qualität und der generellen Beschaffenheit der Gewichtsdaten ab. Auftretende Datenartefakte wie beispielsweise zufällig auftretende Abstände oder Anhäufungen in zur eigentlichen Klassifikation nicht beitragenden Gewichtsbereichen können demnach zu einer Fehlklassifikation führen, da sie als Indikatoren

für Klassengrenzen fehlinterpretiert werden können. Gerade einfache Methoden wie Clustering-Verfahren sind deswegen in Ihrer naiven Form eher ungeeignet.

Besser funktionieren sogenannte Active-Learning Ansätze [79], bei denen es sich formell um eine Hybridvariante aus unüberwachter und überwachter Klassifikation handelt, die aber im Grunde genommen eher den unüberwachten Methoden zuzuordnen wären. Hierbei werden sogenannte positive bzw. negative Keimmengen (Seeds) definiert. Diese enthalten Vergleiche, die zu einer hohen Wahrscheinlichkeit bzw. basierend auf szenariospezifisch definierten Kriterien ausschließlich echte bzw. falsche Übereinstimmungen darstellen. Diese Keimmengen werden dann als Trainingsdaten für die noch unklassifizierten Links verwendet, so dass diese basierend auf Algorithmen wie dem K-Nearest-Neighbour (KNN) oder Support-Vector-Maschinen (SVM) den Keimmengen zugeordnet werden können, bis alle Links schließlich klassifiziert wurden. Peter Christen konnte hierzu in einer Arbeit demonstrieren, dass diese Hybridansätze in der Lage sind, andere unüberwachte Techniken zu übertreffen [71].

Überwachte Klassifikation

Im Gegensatz zur unüberwachten Klassifizierung ist die überwachte Klassifizierung von den Gewichten der Originaldaten unabhängig und basiert auf im Vorfeld spezifizierten Trainingsdaten [41,80]. Hierzu werden Trainingssets benötigt, die in ihrer Beschaffenheit den zu klassifizierenden Daten ähneln und deren echte Übereinstimmungen durch das Teilen derselben ID in beiden Teilsets bekannt sind. Auf diesen Trainingssets lässt sich nun ein Record-Linkage durchführen und basierend auf ausgewählten Qualitätskriterien wie beispielsweise dem F-Measure eine optimale Schranke berechnen. Der Schrankenwert kann nun ebenfalls als Klassifikator für die Originaldaten verwendet werden. Alternativ ist es auch möglich, einen Entscheidungsbaum auf den Trainingsdaten zu generieren, anhand dessen Regeln erzeugt werden können, die die nachfolgende Klassifikation der Originaldaten ermöglichen [81].

Ein Problem dabei ist, dass es im Bereich des Record-Linkage extrem wenige frei-zugängliche auf Realdaten beruhende Trainingssets gibt, die für solch ein Vorgehen geeignet wären. Es existieren zwar einige downloadbare, zur Validierung von Record-Linkage geeignete, Testsets (http://secondstring.sourceforge.net), diese sind aber als Trainingssets in Bezug auf Klassifikation, beispielsweise im medizinischen Bereich, besonders aufgrund abweichender Domäne eher unbrauchbar.

Eine gute Ersatzmöglichkeit kann hierbei die künstliche Erzeugung von Trainingsdaten darstellen. Zum Erzeugen von Patientendaten gibt es sogar eigenständige Software-Kits, wie

z.B. die FEBRL-Toolbox, deren Personengenerierungsmodul auf aus Populationen entnommenen Verteilungswerten beruht [82]. Allerdings waren dem Autor keine Arbeiten bekannt, in denen ein solches Vorgehen, also überwachte Klassifikation auf künstlichen Trainingsdaten, in der Praxis tatsächlich umgesetzt wurde. Das Fehlen festgelegter Standards und der erhöhte Aufwand scheint viele Projektgruppen von überwachter Klassifikation zurückschrecken zu lassen.

An der Johannes-Gutenberg-Universität in Mainz finden Untersuchungen zu neuartigen überwachten bzw. semi-überwachte Klassifikationsmethoden statt [83,84]. Hierbei wird versucht, die Konzepte Bagging und Bumping auf das Szenario des Record-Linkage anzupassen. Bei Bagging und Bumping werden zu zufälligen Ziehungen aus Populationsverteilungen Klassifizierer generiert, deren Mittelwert als finaler Klassifizierer für die Originaldaten zu nutzen ist. Sariyar ist der Meinung, dass die überwachten Methoden dabei die unüberwachten Methoden übertreffen können, allerdings gibt es auch hier noch offene Fragen bezüglich der Parametrisierung, also der genauen Zusammenstellung dieser Trainingsdaten. So stellt zum Beispiel die genaue Festlegung der Anzahl der Trainingsdaten, die beim Bagging bzw. Bumping generiert werden, nach eigenen Angaben ein offenes Problem dar [84].

Regelbasierte Klassifikation

Abseits der unüberwachten bzw. überwachten Klassifizierung existieren auch auf Regeln basierende Klassifikationsmethoden. Zu den Testdaten werden hierbei entweder basierend auf Trainingsdaten oder manuell Regeln konzipiert, die bei Anwendung auf einen Link Auskunft geben, wie wahrscheinlich es sich bei dem Vergleich um einen echten bzw. falschen Link handelt. Solche Regeln bestehen aus Konjunktionen von atomaren Bedingungen wie z.B. *„(ist männlich) UND (Nachname stimmt überein)“.* Das Abarbeiten einer Regel kann im Prüfen neuer Regeln resultieren und es wird gegebenenfalls ein Gewicht vergeben, das zeigt, wie stark die Regel die finale Entscheidung beeinflusst. Nach Abarbeiten aller Regeln wird der Link klassifiziert. Als Struktur solcher abzuarbeitenden Regeln bieten sich Entscheidungsbäume an [41,80].

Hierdurch ist für die Methodik grundsätzlich keine Gewichtsdatei notwendig. Benötigt wird ausschließlich die Information, in welchen Attributen die Einträge übereinstimmen. Unterstützend wurde hierzu eine Variante in Form der Pair-Analysis-Datei in der DKFS verwendet (siehe *Kapitel 1.2.2*).

Bislang (Stand 2012) existiert noch keine ausgiebige vergleichende Prüfung der verschiedenen Klassifikationsmethoden auf verschiedenen Testsets [38].

1.3.4. Softwaresysteme im Bereich des Data-Matchings

Im Bereich des Record-Linkage gibt es eine große Auswahl verschiedener der Thematik zuzuordnenden Softwarepakete. Hierbei handelt es sich um kommerzielle als auch frei zugängliche Pakete. Laut Peter Christen [38] ist es bei den kommerziellen Systemen schwierig, eine übersichtliche Beschreibung der verschiedenen Systeme zur Verfügung zu stellen, da sich diese oftmals nur auf selektierte Teilbereiche der Thematik beschränken. Die Nutzung kommerzieller Systeme ist für die Forschung als kritisch anzusehen, da eine exakte Beschreibung der Algorithmen in der Regel nicht zur Verfügung gestellt wird. Für die Forschung spielen deswegen vor allem Open-Source-Projekte eine wichtige Rolle. Diese werden oft von Forschungseinrichtungen zur Verfügung gestellt und die Algorithmen in assoziierten Publikationen detailliert präsentiert. Im Gegensatz zu kommerziellen Produkten mangelt es hierbei jedoch oft an Usability. *Tabelle 4* gibt eine Übersicht inklusive kurzer Beschreibungen aktueller frei zugänglicher Softwarepakete.

Tabelle 4: Übersicht frei zugänglicher Softwaresysteme im Bereich des Record-Linkage.

System	Beschreibung	Referenz
Big Match	Dient dem Datenabgleich großer Datenmengen. Besitzt jedoch kein User Interface.	[85]
D-Dupe	Ein graphisches Tool dessen Hauptaufgabe die Detektion von Duplikaten in Netzwerken und deren Subnetzwerken ist.	[86]
DuDe	Ein Toolkit bestehend aus mehreren Data-Matching Modulen. Dude besitzt kein grafisches Interface sondern ist als Erweiterung für Javaprojekte konzipiert.	[87]
FEBRL	Beinhaltet Algorithmen zur Datenvorverarbeitung, Deduplikation und dem Data-Matching. Der Fokus liegt hierbei auf der Anwendung für medizinische Datenbanken. Zudem ist es möglich mit FEBRL künstliche Testdaten anhand realer Verteilungswerte zu generieren.	[82]
FRIL	Stark parametrisierbare Data-Matching Software mit graphischem Interface. Teilweise schwierig in der Handhabung.	[88]
Mainzlliste	Webbasierter Pseudonymisierungsdienst inklusive gewichtsbasiertem, modularem Record-Linkage System.	bitbucket.org/ medinfo_mainz /mainzlliste/
Merge ToolBox	Umfangreiches Data-Matching Paket, das die Anwendung von Privacy-Preserving-Record-Linkage mittels Bloom-Filtern gestattet. Die Module bauen teilweise auf der kommerziellen Software Stata auf.	[89]

System	Beschreibung	Referenz
OYSTER	Wurde zur Erfassung und Verwaltung von Studentenakten erstellt. Enthält unter anderem Module für probabilistisches Record-Linkage.	[90]
R RecordLinkage	Paket für probabilistisches Record-Linkage für die Statistiksoftware „R“.	[91]
SILK	Umfangreiches Data-Matching System, das Daten im RDF Format speichert und abgleicht.	[92]
Sim Metrics	Beinhaltet eine große Auswahl approximativer Textvergleichs-Funktionen.	sourceforge.net /projects/simmetrics
TAILOR	Umfangreiches Toolkit zu verschiedenen Anwendungen aus dem Bereich des Record-Linkage inklusive einiger Klassifikationsmethoden.	[93]
WHIRL	Beinhaltet einen regelbasierten Klassifikationsansatz.	[94]

1.3.5. **Möglichkeiten der Evaluation**

Das Hauptanliegen beim Datenabgleich ist das Erzielen einer möglichst hohen Abgleichs Qualität, durch die sich gleichzeitig die Güte von verschiedenen methodischen Ansätzen abschätzen und vergleichen lässt. Diese lässt sich anhand der Anzahl von echt bzw. falsch

		Realität	
		Übereinstimmung (MATCH)	***Nicht-Übereinstimmung (NON-MATCH)***
Klassifikation	***Echter Link (LINK)***	**Echt Positive (TRUE POSITIVES)**	**Falsch Positive (FALSE POSITIVES)**
	Falscher Link (NON-LINK)	**Falsch Negative (FALSE NEGATIVES)**	**Echt Negative (TRUE NEGATIVES)**

Abbildung 8: Kontingenztafel mit dem Urteil der Klassifikation und der tatsächlichen Klasse.

ermittelten Übereinstimmungen, bzw. echt bzw. falsch ermittelten Nicht-Übereinstimmungen berechnen. Die vier beschriebenen Beobachtungen lassen sich übersichtlich in einer vier Felder Tafel, (*siehe Abbildung 8*) auf das Szenario des Record-Linkage angepasst, darstellen [33,95].

Durch die in der Vier-Felder Tafel aufgelisteten statistischen Maßeinheiten (True Positives (TP), False-Positives (FP), False-Negatives (FN), True-Negatives (TN)) lassen sich verschiedene Qualitätsmaße berechnen. Als häufig in der Statistik verwendete Qualitätsmaße wären hierzu die Spezifität sowie die Sensitivität zu nennen (siehe *Formel 11,12*):

$$Spezifität = \frac{TN}{TN + FP} \qquad (11)$$

$$Sensitivität = \frac{TP}{TP + FN} \qquad (12)$$

Die Spezifität berechnet den Anteil von Vergleichen, die als falsche Links klassifiziert wurden und bei denen es sich tatsächlich um Nicht-Übereinstimmungen handelt. Die Sensitivität berechnet den Anteil von Vergleichen von echten Übereinstimmungen an der Menge der vorhergesagten echten Links. Für das Prüfen von Methoden im Bereich des Record-Linkage, wie beispielsweise die Prüfung der Performanz verschiedener Klassifikatoren, zeigt sich, dass der Spezifität im Regelfall eher niedrigere Wichtigkeit zugeordnet werden sollte [71]. Der Grund hierfür ist, dass abhängig von den Blocking-Variablen, beim Record-Linkage in der Praxis, vor allem bei den Vergleichen von Nicht-Übereinstimmungen, Gewichte berechnet werden müssen. Das Produkt der Datensetgrößen ist hierbei der Maximalwert der Vergleiche, bei denen es sich in der Regel nur zum kleinsten Teil um echte Übereinstimmungen handelt. Bei dem Großteil der Daten wird es sich also bei ansatzweiser korrekter Klassifikation um True-Negatives, also Nicht-Übereinstimmungen, die als falsche Links klassifiziert wurden, handeln. Durch die hohe Zahl der True-Negatives im Vergleich zu auftretenden False-Positives werden in den meisten Szenarien auch bei oftmals stark variabler Positionierung eines Klassifikators Spezifitätswerte um 99% erzielt. Eine Ausrichtung eines Klassifikators an der maximalen Sensitivität hingegen kann zur Nicht-Berücksichtigung vieler echter Übereinstimmungen führen.

Ein geeigneteres Qualitätsmaß im Kontext des Record-Linkage stellt deshalb der F-Measure-Wert da [71,96]. Hierbei handelt es sich um den harmonischen Mittelwert der Sensitivität und des positiv prädiktiven Wertes (siehe *Formel 13, 14*).

$$PPV = \frac{TP}{TP + FP} \quad (13)$$

$$FM = 2 * \frac{PPV * Sensitivität}{PPV + Sensitivität} \quad (14)$$

Beim positiv prädiktiven Wert (PPV) handelt es sich um den Anteil der korrekt klassifizierten, echten Übereinstimmungen an der Menge aller echten Übereinstimmungen. Im Bereich des Record-Linkage wäre also ein hoher F-Measure-Wert mit einer hohen Abgleichsqualität zu interpretieren. Die Bestimmung der Qualitätsmerkmale ist nur dann möglich, wenn die echten Übereinstimmungen bekannt sind und sich die finale Klassifikation mit den tatsächlichen Gegebenheiten abstimmen lässt. Hierdurch ist die Qualität des Record-Linkage nur in Tests, nicht aber im Realeinsatz berechenbar. Tests, bei denen die Übereinstimmungen bekannt sind, bezeichnet man auch als Gold-Standard [97]. Realdaten, zu denen eine Goldstandardanalyse möglich ist, sind jedoch im Bereich des Record-Linkage extrem selten und es existieren hierzu nur wenige Arbeiten [19].

1.4. **Zielsetzung**

Anhand einer Studie zu familiärem Darmkrebs (siehe *Kapitel 1.2.1*) wurden im Bereich des Record-Linkage Unsicherheiten bei der manuellen, binären Klassifikation, die zu einer Verminderung der Abgleichsqualität führen könnten, erkannt (siehe *Kapitel 1.2.2*). Unterstützend, oder auch alternativ, existieren bereits verschiedene automatisierte Klassifikationsansätze, nennenswert sowohl unüberwachte als auch überwachte Klassifikationssysteme (siehe *Kapitel 1.3.3*). Gerade zu überwachter Klassifikation existieren jedoch im Moment keine klaren Standards. Auch werden dort zusätzlich zu den Originaldaten Trainingsdaten benötigt.

Da reale Trainingsdaten meist nicht zur Verfügung stehen, könnten alternativ künstliche Trainingsdaten eingesetzt werden. Zu deren konkreter Beschaffenheit fanden sich jedoch keine Empfehlungen. Ausgangspunkt der Arbeit war die Überlegung, künstliche Trainingsdaten zu erzeugen, die den Originaldaten in hohem Maße ähneln. Basierend auf dieser Überlegung ergab sich die Zielsetzung, die optimale Parametrisierung bei der Konstruktion von künstlichen Trainingsdaten bei der überwachten Klassifizierung zu untersuchen und darauf aufbauend Empfehlungen zu erarbeiten.

Weiterhin fehlten Informationen und umfangreiche vergleichende Tests zur Performanz unüberwachter sowie überwachter Methoden im direkten Vergleich [38]. Das zu erarbeitende

überwachte Klassifikationssystem sollte deswegen mit verschiedenen, unüberwachten Klassifikationsansätzen sowie der manuellen Schrankengebung, wie sie in der DKFS Anwendung findet, verglichen werden.

Bei den zu vergleichenden unüberwachten Methoden sollte es sich sowohl um eine einfache Clustering-Methode, als auch um eine fortgeschrittene Technik aus dem Bereich des Active-Learnings, die anderen unüberwachten Methoden qualitativ überlegen ist, handeln [71].

Die Testdaten sollten sich in spezifizierten Parametern, der Größe, dem Überlappungsbereich, sowie der Fehlerhäufigkeit unterscheiden.

2. Material und Methoden

2.1. Vorbereitende Arbeiten und Arbeitsmaterial

2.1.1. Verwaltung der Arbeitsumgebung

Für die angestrebten Analysen der gegebenen Arbeit waren aufwendige Berechnungen und Arbeitsschritte notwendig, die manuell nicht mehr im realen Zeitrahmen zu bewältigen gewesen wären. Hierdurch bestand die Notwendigkeit fortgeschrittener Programmiertechniken. Als zugrunde liegende Programmiersprache der implementierten Programme fand Java 1.7 Verwendung – als Programmierinterface hierzu die Software Eclipse (https://www.eclipse.org/).

Die Programme selber wurden kursiv und durch einen in spitzen Klammern nachfolgenden Index entsprechend *Kapitel 7 – Anhang E* im Text aufgeführt. Die Erstellung der in dieser Arbeit dargestellten Plots und einiger mathematischer Auswertungen erfolgte über die Statistik Software „R“ (http://www.r-project.org/).

Ein Abbild der finalen Arbeitsumgebung, also aller erzeugten Programme bzw. Klassen und Daten, wurde zur nachhaltigen Speicherung vom Autor dieser Arbeit gesichert und aufbewahrt. Für die teilweise zeitintensiven Berechnungen war ein leistungsstarker Rechner notwendig. *Tabelle 5* skizziert die wichtigsten Hardwarekennziffern des zumeist verwendeten Systems.

Tabelle 5: Wichtigste Hardwarekomponenten des Arbeitssystems.

Prozessor	**Arbeitsspeicher**
Intel(R) Core™ i7-3770 CPU @3,4 GHz	8 GB-RAM

2.1.2. Record-Linkage: Spezifikation und Implementierung

Für die zugrunde liegenden Tests und Entwicklungen wurde eine leicht abgewandelte Variante des probabilistischen Privacy-Preserving-Record-Linkage, das auch in der Familienstudie Anwendung fand, verwendet [67]. Hierbei handelte es sich um eine Implementierung des Fellegi und Sunther Algorithmus nach Spezifikation von Martin Meyer [31,39]. Die konkrete

Implementierung wurde innerhalb des Programmes *RecordLinkage<1>,* sowie der assoziierten Klasse *RecordLinkageInput<2>* umgesetzt.

Als Input dienten diesem System jeweils zwei Datensätze, die bereits standardisierte, einwegverschlüsselte Kontrollnummern von identifizierenden Daten (IDAT) beinhalteten. Das Format dieser Daten musste dem Rückgabeformat des Programmes *GenerateControlNumbers<6>* entsprechen, das zugrunde liegende Personendaten gemäß Regelvorgaben aus UNICON [55] (siehe *Kapitel 1.3.2*) erst standardisiert und dann mithilfe der Hash-Funktion SHA-2 (256-Bit) [62] einwegverschlüsselt.

Tabelle 6 beschreibt die in dieser Arbeit genutzten identifizierenden Basisdaten wie auch die hierauf basierenden standardisierten, einwegverschlüsselten Kontrollnummern so wie sie von der Klasse GenerateControlNumbers<6> erzeugt werden.

Tabelle 6: In dieser Arbeit zur Gewichtsberechnung genutzte IDAT.

IDAT	**Segmentierung in Kontrollnummern.**
Nachname	NACHNAME1, NACHNAME2, NACHNAME3
Vorname	VORNAME1, VORNAME2, VORNAME3,
Geburtsdatum	GEBURTSTAG, GEBURTSMONAT, GEBURTSJAHR
PLZ	PLZ
Wohnort	ORT
Geschlecht	GESCHLECHT
Personen-Identifikationsnummer	PID

Während des Standardisierungsschrittes wurdenzudem eine Reihe von Kontrollnummern, die ausschließlich als Blocking-Variablen dienten, erzeugt. Hierbei handelte es sich um den phonetischen Nachnamen, den phonetischen Vornamen sowie das Geburtsdatum (siehe *Tabelle 7*).

Tabelle 7: Blocking-Variablen inklusive der IDAT, aus der die BV generiert wurden.

IDAT	**Blocking-Variablen**
Nachname	PHO_NACHNAME
Vorname	PHO_VORNAME
Geburtsdatum	GEBURTSDATUM

Bei diesen Variablen wurde auf eine Segmentierung während der Standardisierung verzichtet. Vorname und Nachname wurden anhand der Kölner Phonetik in ihre entsprechende phonetische Variante generalisiert [57]. Der Algorithmus zur Kölner Phonetik stammt aus einer von Apache zur Verfügung gestellten externen Programmier-Bibliothek (http://commons.apache.org/proper/commons-codec/). Während bei der Umsetzung der Familienstudie das Geburtsjahr als Blocking-Variable verwendet wurde, fiel in dieser Arbeit die Wahl auf das Geburtsdatum, da das Geburtsjahr eine starke Generalisierung darstellt und durch die Verwendung des spezifischeren Geburtsdatums wesentlich weniger Übereinstimmungen in der konkreten Blocking-Variable und demnach nachfolgende Gewichtsberechnungen erzeugt wurden. Diese Maßnahme erschien aufgrund der vielen kommenden Auswertungen, in Hinblick auf realisierbare Performanz, notwendig. Beim Blocking handelte es sich um Standard-Blocking (siehe *Kapitel 1.3.2*) auf den drei genannten Blocking-Variablen. Potentielle Links wurden nur einmalig abgespeichert, auch wenn diese in mehreren Variablen übereinstimmten [67].

Zu den potentiellen Links fand eine Gewichtsbestimmung statt. Bei dieser wurde ein durch Fellegi und Sunther [31] konzipierter Ansatz verwendet. Hierbei werden vom Typ her gleiche Kontrollnummern eines potentiellen Links (also beispielsweise das Geschlecht zweier Personen) abgeglichen und Einzelgewichte berechnet. Die Höhe dieses Einzelgewichtes basiert auf den Häufigkeiten der verglichenen Ausprägungen (u-Wert) und den in dieser Variable erwarteten invertierten Fehlerhäufigkeiten (m-Wert). Siehe *Kapitel 1.3.3* für exaktere Erläuterungen. Die Einzelgewichte wurden nachfolgend zu Gesamtgewichten aufaddiert.

Um zu gewährleisten, dass bei unterschiedlich auftretender Reihenfolge von Attributswerten in den zugrunde liegenden IDAT, wie beispielsweise innerhalb von Doppelnamen, Übereinstimmungen zu erkennen sind (z.B. Müller-Wagner/Wagner-Müller), wurden Matching-Arrays für Vornamen (VORNAME1, VORNAME2, VORNAME3) und Nachnamen (NACHNAME1, NACHNAME2, NACHNAME3) verwendet, in denen jeweils alle enthaltenen Kontrollnummerausprägungen paarweise im Kreuzprodukt miteinander abgeglichen wurden. Zuerst wurden hierbei die Kontrollnummern auf paarweise Übereinstimmungen untersucht. Beim Auffinden von Übereinstimmungen wurde ein Einzelgewicht berechnet und die konkreten Kontrollnummerausprägungen wurden aus dem jeweiligen Matching-Array entfernt bis nur noch Nicht-Übereinstimmungen oder überhaupt keine Werte mehr übrig waren. Anschließend wurden Gewichte zu den verbleibenden Nicht-Übereinstimmungen berechnet.

Die Gewichtung der potentiellen Links wurde in eine Gewichtsdatei geschrieben. Jeder potentielle Link belegte hierbei eine Zeile und bestand aus den PIDs der verglichenen Einträge

der verschiedenen Datensätze, sowie deren Übereinstimmungsgewicht. *Abbildung 9* illustriert den schematischen Ablauf des in der Arbeit verwendeten Record-Linkage-Systems.

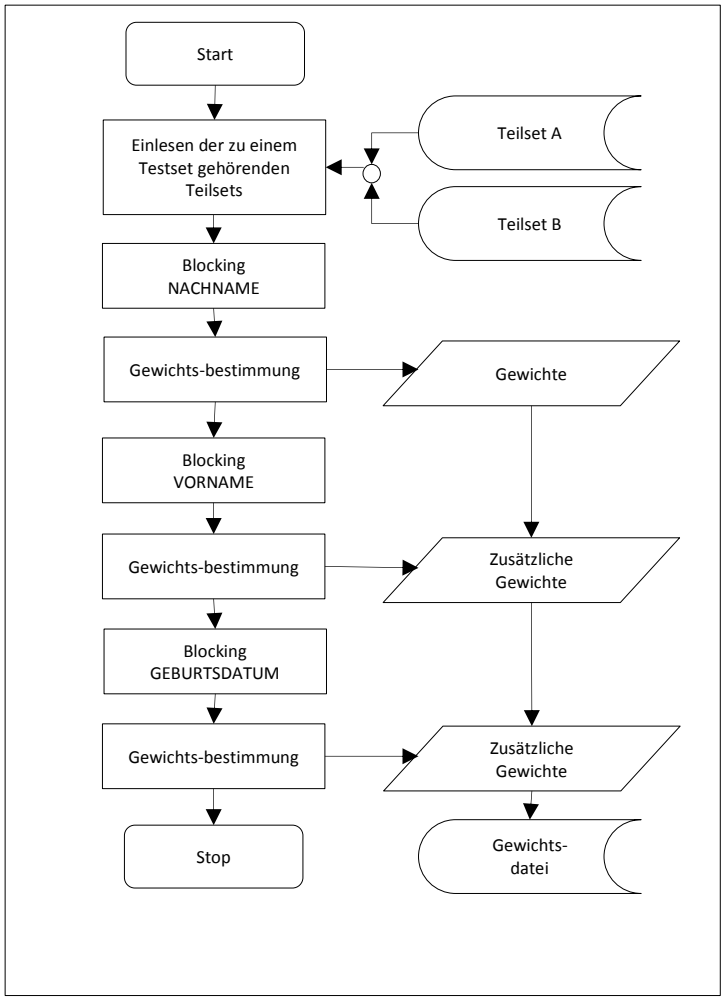


Abbildung 9: Schematischer Ablauf des für diese Arbeit verwendeten Record-Linkage-Systems.

2.1.3. **Beschreibung der verwendeten klinischen Daten**

Für diese Arbeit wurde ein realer Datensatz, bestehend aus Personen identifizierenden Daten zu 46.629 Patienten des Klinikums Großhadern (http://www.klinikum.uni-muenchen.de) verwendet. Die Patientendaten wurden dabei im Vorfeld anhand des Programmes *GenerateControlNumber<6>* standardisiert und einwegverschlüsselt. Es handelte sich dabei um eine zufällige Stichprobe aus einer Gesamtmenge von insgesamt 466.286 Patienten, die in den Jahren 2008-2012 im Klinikum zur Behandlung registriert wurden (dieser Datensatz enthielt keine Daten von Patienten, deren Aufnahme storniert wurde). Der zur Verfügung gestellte

Datensatz entsprach somit einem Anteil von ca. 10% der Patienten, die während des genannten Zeitraumes tatsächlich behandelt wurden.

Durch die Größe des Datensatzes sollte eine relativ bevölkerungsnahe und realistische Verteilung von Attributen wie beispielsweise Vornamen oder Nachnamen in der Region zu erwarten sein. Dadurch, dass die meisten Patienten spekulativ aus dem Großraum München und Umgebung stammen sollten, war zu erwarten, dass der Datensatz im Gegensatz zu komplett künstlichen Datensätzen zudem interessante Verwandtschaftsbeziehungen wie etwa das Vorkommen von Zwillingen enthielt, die in der Regel hohe Anforderungen an ein Record-Linkage stellen.

2.2. **Überwachte Klassifizierung – angestrebtes Vorgehen**

Im Zuge dieser Arbeit galt es unter anderem, ein überwachtes Klassifizierungssystem zu entwickeln und mit unüberwachten Klassifikationstechniken abzugleichen. Dieses überwachte System sollte dabei, angepasst an die Originaldaten, Trainingssets konstruieren auf denen ein optimaler Trainingsset-spezifischer Klassifikator ermittelbar wäre welcher schließlich als Klassifikator auf den Originaldaten verwendet werden könnte. Die genaue Konstruktion der Trainingssets in Bezug auf die einzelnen Konstruktionsparameter wie beispielsweise die Größe der Teilsets sollte innerhalb dieser Arbeit ermittelt, und auf beste Performanz (Abgleichsgüte) hin optimiert werden (*siehe Kapitel 2.5 bzw. 2.6*). Der generelle Ablauf der angestrebten überwachten Klassifizierungsmethodik konnte aber bereits spezifiziert werden und unterteilte sich in folgende Schritte (siehe auch *Abbildung 10*):

1. Bilden von N Trainingssets A und B, basierend auf den abzugleichenden originalen Datensätzen A und B nach Konstruktions-Verfahren X (Details zu X galt es zu erarbeiten). N richtet sich hierbei nach der Performanz des zugrunde liegenden Hardwaresystems, wobei ein hoher Wert den maximal möglichen Fehler verringert.
2. Auf den erzeugten N Trainingssets wird ein Record-Linkage durchgeführt.
3. Bestimmung des optimalen Klassifikators auf jedem der erzeugten N Trainingssets. Die optimale Schranke wird hierbei durch nachvollziehbare Übereinstimmungen (gleiche IDs) innerhalb des Überlappungsbereiches der Trainingsdaten und dem hieraus berechenbaren F-Measure-Wert berechnet.
4. Zu den ermittelten N Trainingsset spezifischen Klassifikatoren wird ein neuer Klassifikator, der das arithmetische Mittel der einzelnen Klassifikatoren darstellt, berechnet. Dieser neue Wert dient als Klassifikator für die Originaldaten.

5. Die Originaldaten werden per Record-Linkage abgeglichen.
6. Der in (4) berechnete Klassifikator dient als unüberwachter Klassifikator auf den Originaldaten.

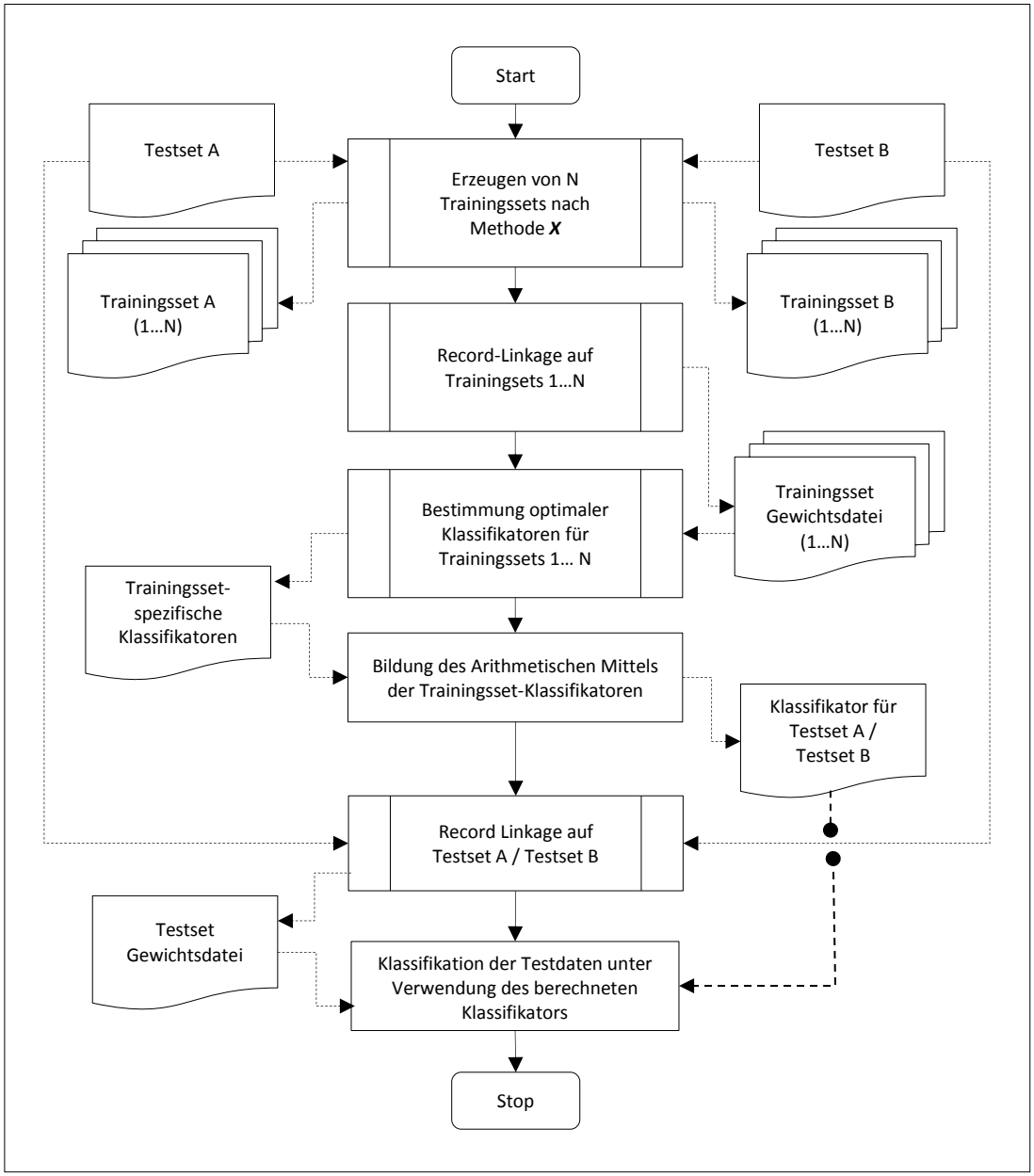


Abbildung 10: Konzept zur angestrebten überwachten Klassifizierungsmethodik.

Zur Entwicklung und Parameteroptimierung des Konstruktionsverfahrens X wurden in dieser Arbeit umfangreiche Tests und Performanzvergleiche bezüglich der Abgleichsgüte benötigt. Diese sollten anhand einer Vielzahl von Testsets, basierend auf den unter Kapitel 2.1.3 beschriebenen klinischen Daten erstellt werden.

2.3. Erzeugung von Testsets anhand klinischer Daten

2.3.1. Notwendigkeit der Testset-Erzeugung

Zur Einschätzung bestehender als auch neu entwickelter Klassifizierungsmethoden waren Datensätze notwendig, anhand derer sich Gütekriterien quantifizieren ließen und somit einen Vergleich der verschiedenen Methoden ermöglichten. Solche Datensätze werden im Bereich des maschinellen Lernens auch als Testsets bezeichnet [80]. Es war davon auszugehen, dass je nach Beschaffenheit der Testsets unterschiedliche Klassifizierungsmethoden zu verschieden guten Ergebnissen führen würden. Aus diesem Grund war es ratsam, eine möglichst breite Palette an Testsets mit verschieden Charakteristiken als Datengrundlage für Analysen zu verwenden. Im Bereich des medizinischen Record-Linkage ist die Anzahl an offen zugänglichen, geeigneten Testsets jedoch beschränkt oder vom Kontext her unpassend. Das Problem liegt hierbei nicht grundsätzlich im Zugang zu Patientendaten an sich, sondern in der notwendigen Beschaffenheit der Testsets. Ein geeignetes Testset hat aus jeweils zwei Datenmengen zu bestehen, die eine gemeinsame Teilmenge besitzen. Diese gemeinsame Teilmenge muss bekannt und über gemeinsame IDs oder andere Schlüsselelemente eindeutig zueinander zuordenbar sein (siehe *Abbildung 11*).

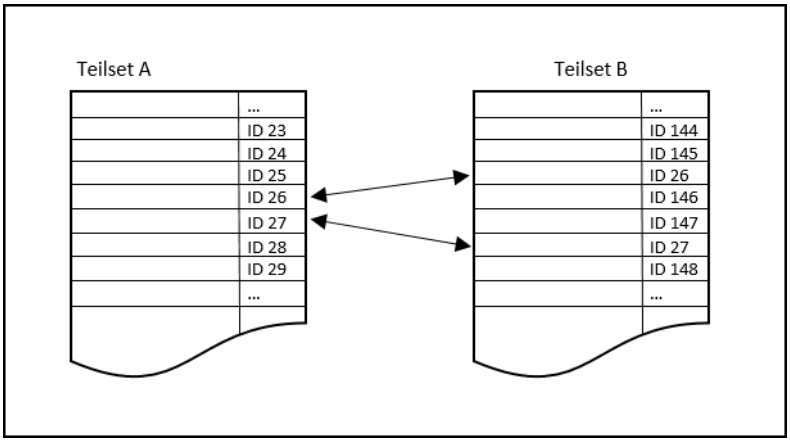


Abbildung 11: Darstellung eines für im Kontext des Record-Linkage nutzbaren Testsets.

Man bezeichnet diese Teilmenge auch als Menge der echten Übereinstimmungen (Matches). Im weiteren Verlauf der Arbeit wurden die größeren der beiden Teilsets eines Datensets jeweils als Teilset A und die kleineren der Teilsets als Teilset B bezeichnet. Nach einem Record-Linkage-Durchlauf ließ sich die Güte des Record-Linkage anhand der Diskrepanz der Übereinstimmungen und Nicht-Übereinstimmungen zu den als echt bzw. falsch klassifizierten Links berechnen (siehe *Kapitel 1.3.5*).

Den Güteberechnungen lag die Annahme zu Grunde, dass die echten Übereinstimmungen in den Testsets korrekt zueinander zugeordnet wurden. Auf Realdaten gibt es hierzu keine Garantie, allerdings spricht man von einem Goldstandard, wenn die Übereinstimmungen im Klartext manuell kontrolliert zueinander zugeordnet werden [19]. Im Kontext von Patientendaten, die beim Zusammenführen aus verschiedenen Quellinstitutionen eine Einwegverschlüsselung benötigen, ist eine solche Zuordnung im Klartext, und somit die Erzeugung eines dem Goldstandard entsprechenden Datensatzes in der Regel nicht oder nur unter speziellen Bedingungen (beispielsweise innerhalb einer Kohorte) möglich. Diese Arbeit strebte Analysen auf einem umfangreichen Set verschieden zusammengesetzter Testsets an. Hierdurch wurden Methoden benötigt, die die Konstruktion solcher Testsets erlaubten.

2.3.2. Spezifizierung der Parameter zur Testset-Erzeugung

Zu den in dieser Arbeit durchgeführten Untersuchungen sollten die gegebenen Patientendaten des Klinikums (siehe *Kapitel 2.1.3*) genutzt werden, um eine Reihe von künstlichen, jedoch auf Realdaten basierenden Testsets zu erstellen. Für eine umfangreiche Auswahl an Testsets wurden interessante und passende Charakteristiken spezifiziert, anhand deren Kombination die verschiedenen Testsets letztendlich erstellt werden sollten. Bei den spezifizierten Charakteristiken handelte es sich um die Größe der Teilsets, die Größe des Überlappungsbereiches, also der Teilmenge von Patienten mit gleicher ID in beiden Teilsets, sowie die individuell auftretenden Fehlerraten zwischen den Attributen der Patienten innerhalb des Überlappungsbereiches (siehe *Tabelle 9*). Zur Vereinfachung wurde die Häufigkeit des Auftretens von Fehlern im Überlappungsbereich auch als Beschaffenheit oder Qualitätsstufe des jeweiligen Testsets bezeichnet. Ähnliche Charakteristiken werden bereits in Arbeiten von Peter Christen zur Erzeugung künstlicher Testsets verwendet [71]. Jede Charakteristik besaß mögliche Ausprägungen wie in *Tabelle 8* weiter spezifiziert. Dabei handelte es sich um die mögliche Anzahl von Patienten pro Teilset (Größe), die Anzahl von identischen Patienten in beiden Teilsets (Überlappung) sowie die Qualitätsstufe. Eine Qualitätsstufe von 1 beschrieb eine gute Datenqualität d.h. ein geringes Auftreten von Fehlern in Attributswerten von Patienten im Überlappungsbereich, wohingegen der Wert 10 den schlechtesten Wert, also ein häufiges Auftreten von Fehlern, darstellte. Anzumerken ist, dass Größenanordnung, (also [100:1000] bzw. [1000:100]) der Teilsets für diese Arbeit keine Rolle spielte, wodurch sich die hieraus ergebenden Kombinationen auf 10 beschränkten. Insgesamt konnten somit 400 Testsets mit einzigartiger Kombination von Charakteristiken erzeugt werden (siehe *Formel 15*).

Tabelle 8: Ausprägungsliste der Konstruktionsparameter.

Größe	Überlappung	Qualitätsstufe
100	5%	1-10
1000	25%	
10000	50%	
25000	75%	

$$|Testsets| = \frac{(|Größe| + |Teilsets| - 1)!}{(|Größe| - 1)!\,|Teilsets|!} \times |Überlappung| \times |Beschaffenheit| \quad (15)$$
$$= \frac{(4 + 2 - 1)!}{(4 - 1)!\,2!} \times 4 \times 10 = \frac{5!}{3! \times 2!} \times 4 \times 10 = 10 \times 4 \times 10 = 400$$

Durch die hohe Anzahl an Testsets deckte die Arbeit somit eine sehr breite Palette von Szenarien bzw. Datenbeständen ab, die ähnlich auch in der Realität auftreten könnten. *Abbildung 12* zeigt hierbei den schematischen Ablauf der Automatisierung der Testset-Erzeugung. Diese wurde mithilfe des Programmes *CreateTestsets*<7> umgesetzt. Die 400 Testsets, jeweils bestehend aus einem Teilset A, bzw. einem Teilset B, belegten insgesamt 5,04 GB Speicherplatz.

Für den weiteren Verlauf der Arbeit war es wichtig, die Kenntnis zur genutzten Parametrisierung der Testdaten zu dokumentieren. Dies geschah direkt über den Dateinamen (siehe *Abbildung 13*).

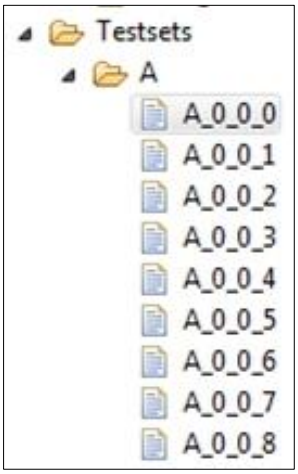


Abbildung 12: Ausschnitt aus dem Projektverzeichnis der Programmierumgebung.

Ein führender Großbuchstabe beschrieb dabei das Teilset (A bzw. B), gefolgt von durch Unterstrich separierten Parameterwerten. Der erste numerische Wert hierbei kodierte die Größenkombination, der zweite Wert die Überlappung und der dritte Wert die Beschaffenheit.

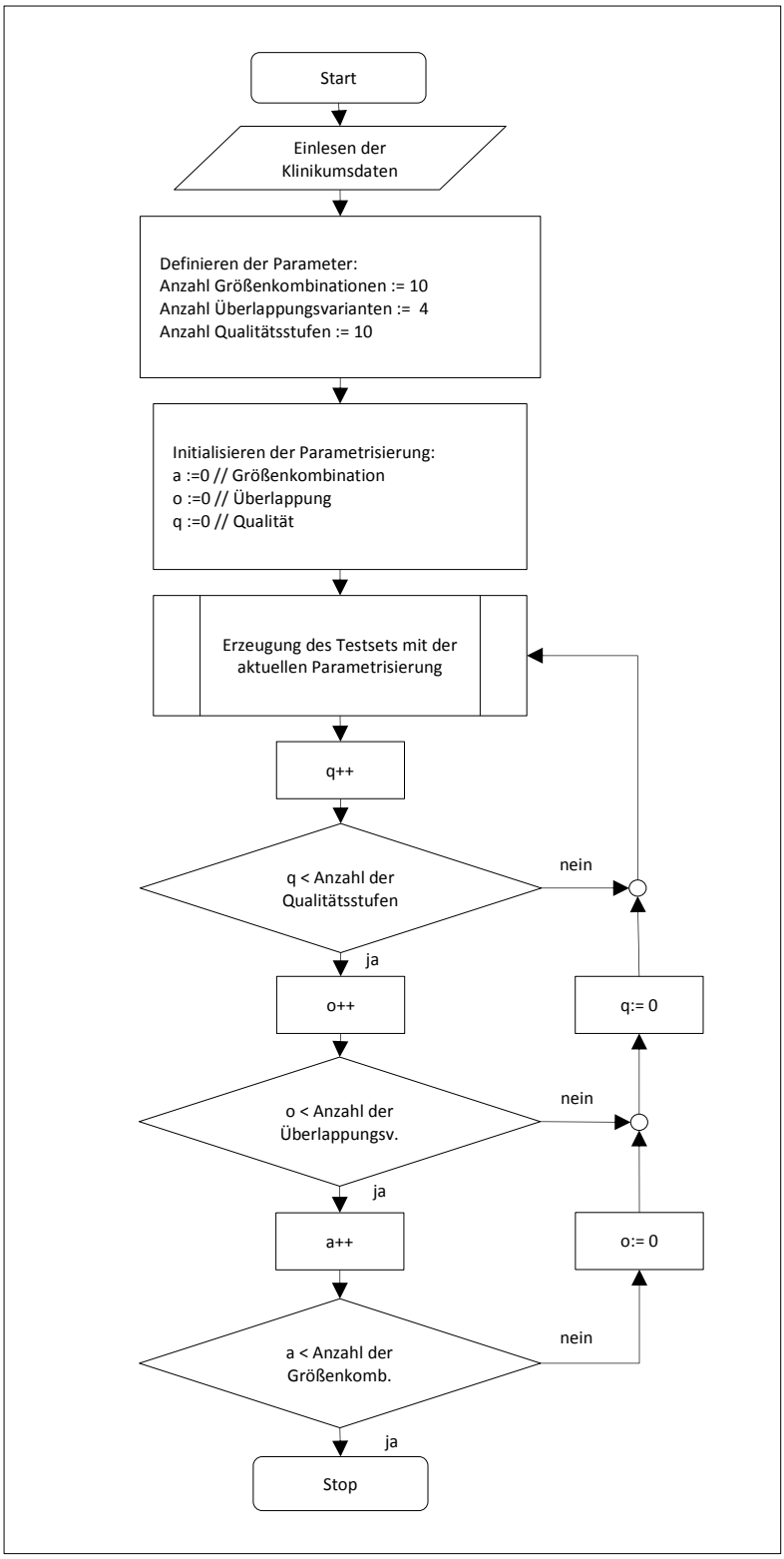


Abbildung 13: Automatisierter Ablauf der Testset-Erzeugung.

Die numerischen Werte standen hierbei stellvertretend für die in *Tabelle 9* beschriebenen Ausprägungen.

Tabelle 9: Kodierung der Testset-Benennung. (siehe Abbildung 13)

Größenkombination		Überlappung		Beschaffenheit	
Vermerk im Dateiname	**Wert**	**Vermerk im Dateiname**	**Wert**	**Vermerk im Dateiname**	**Wert**
0	[100:100]	0	5%	0	Q1
1	[100:1000]	1	25%	1	Q2
2	[100:10000]	2	50%	2	Q3
3	[100:20000]	3	75%	3	Q4
4	[1000:1000]			4	Q5
5	[1000:10000]			5	Q6
6	[1000:20000]			6	Q7
7	[10000:10000]			7	Q8
8	[10000:20000]			8	Q9
9	[20000:20000]			9	Q10

2.3.3. **Konkrete Implementierung der Testset-Erzeugung**

Erzeugung von Teilset A

Bei der Erzeugung der individuellen Testsets zu dieser Arbeit wurde wie folgend vorgegangen. Aus dem Basisdatensatz des Klinikums wurden Daten entsprechend der Größe des zu erstellenden größeren Teilsets, basierend auf dem für das Testset zugeordneten Größenparameter, gezogen (siehe *Abbildung 14a*). Es handelte sich hierbei um Ziehen ohne Zurücklegen, weswegen in diesen neu erstellten Teilsets jeweils keine Patienten doppelt vorkamen (unter der Annahme, dass die Basisdaten des Klinikums weitestgehend duplikatfrei sind). Weiterführend wurden die jeweils größeren Teilsets eines Testsets als Testset A bezeichnet.

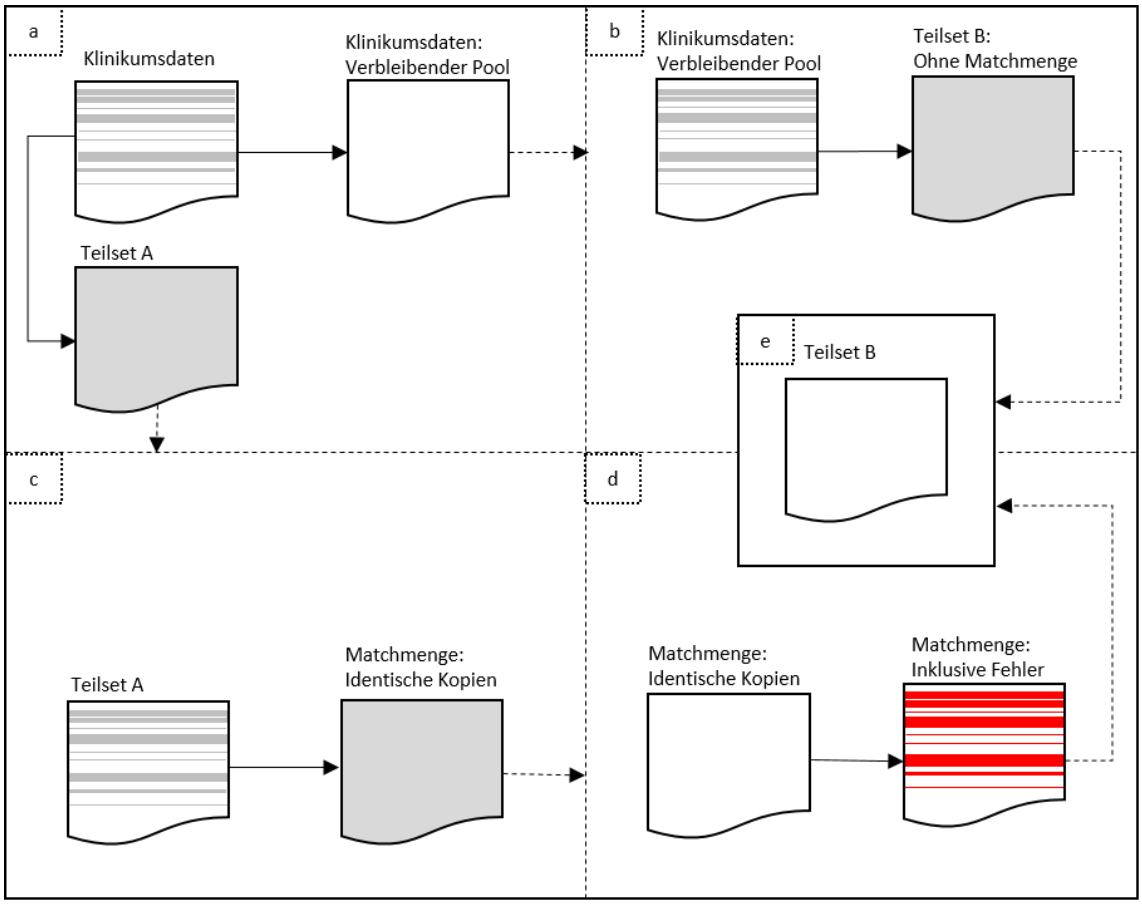


Abbildung 14: Erzeugung individueller Testsets basierend auf unterschiedlicher Parametrisierung.

Erzeugung von Teilset B

Auffüllen des Teilsets abzüglich des Überlappungsbereiches

Die Erstellung der kleineren Teilsets, die weiterführend jeweils als Teilset B bezeichnet wurden, stellte sich als etwas komplexer dar. Das jeweilige Teilset B wurde gemäß des gegebenen Größenparameters aus demselben Topf an noch verbleibenden Klinikumsdaten, aus dem bereits Teilset A erstellt wurde, mit Patienten aufgefüllt. Zu beachten war allerdings, dass der Anteil der Überlappung in diesem Teilset B zu diesem Zeitpunkt noch nicht belegt wurde (siehe *Abbildung 14b*).

Erstellen des Überlappungsbereiches

Zu dem noch nicht befüllten Überlappungsbereich wurden nun Patienten (ohne Duplikate) aus dem Teilset A in das Teilset B kopiert. Der Überlappungsbereich enthielt somit die Patienten, die sowohl in Teilset A als auch in Teilset B auftraten und die über die gleich bleibende PID in beiden Datensätzen erkennbar waren (siehe *Abbildung 14c*).

Ohne weitere Bearbeitung wäre dieser Überlappungsbereich nun durch ein Record-Linkage problemlos zu identifizieren gewesen, da es sich um direkte Kopien, also 100%ige Übereinstimmungen in den Attributen zwischen den Patienten der beiden Teilsets handelte. Die Testsets dienten jedoch dem Zweck, realistische Szenarien so gut wie möglich zu simulieren. Aus diesem Grund wurden die Attribute der Patienten im Überlappungsbereich gemäß dem Beschaffenheitsparameter des jeweiligen Testsets verunreinigt bzw. mit Fehlern versehen.

Einfügen von Fehlern in Kontrollnummern der Patienten innerhalb des Überlappungsbereiches

Während dieses Schrittes wurden Fehler entsprechend der durch die einzelnen Beschaffenheitsstufen (1 bis 10) definierten Fehlerhäufigkeiten in die Kontrollnummern der Patienten im Überlappungsbereich übertragen. Die verwendeten Fehlerhäufigkeiten leiteten sich hierbei aus zwei Berichten ab, zum einem aus einen Bericht aus dem Krebsregister NRW [73], zum anderen zu generell empfohlenen Schätzwerten der m-Werte während eines Record-Linkage (also den invertierten Fehlerhäufigkeiten) in Krebsregistern [39]. Anhand der beiden Referenzen wurden hierbei die Beschaffenheitsstufen 1 bzw. 2 erstellt, die eine gute Datenqualität, so wie sie in gepflegten Registern vorkommen sollte, darstellen sollten. Die Differenz in den attributabhängigen Fehlerwahrscheinlichkeiten zwischen Beschaffenheitsstufe 1 und Beschaffenheitsstufe 2 wurde verwendet, um die Fehlerhäufigkeiten in den restlichen Beschaffenheitsstufen (3-10) zu ermitteln. Die Beschaffenheitsstufe 10 stellte somit Testsets mit der niedrigsten Datenqualität dar. Die genauen Fehlerhäufigkeiten, abhängig von der Beschaffenheitsstufe, werden in *Tabelle 10* bzw. *Abbildung 15* wiedergegeben.

Tabelle 10: Fehlerhäufigkeiten abhängig von Qualitätsstufe und Attributsgruppe

Attributsgruppe	**Konkrete Attribute**	**Fehlerquote nach Beschaffenheit**	
		Start	**Faktor**
Namensattribute	NACHNAME1,NACHNAME2,NACHNAME3, VORNAME1,VORNAME2,VORNAME3	0,025	0,025
Datumsangaben	GEBURTSTAG,GEBURTSMONAT, GEBURTSJAHR	0,01	0,01
Adressangaben	PLZ,ORT	0,05	0,05
Geschlecht	GESCHLECHT	0,001*	0,005

Der Startwert gibt die initialen Fehlerhäufigkeiten in den einzelnen Attributsgruppen bei einer Beschaffenheitsstufe 1 wieder. Für jede Beschaffenheitsstufe erhöhte sich die Fehlerhäufigkeit um einen attributsspezifischen Faktor, der, wie erwähnt, der Differenz aus Q1 und Q2 entsprach.

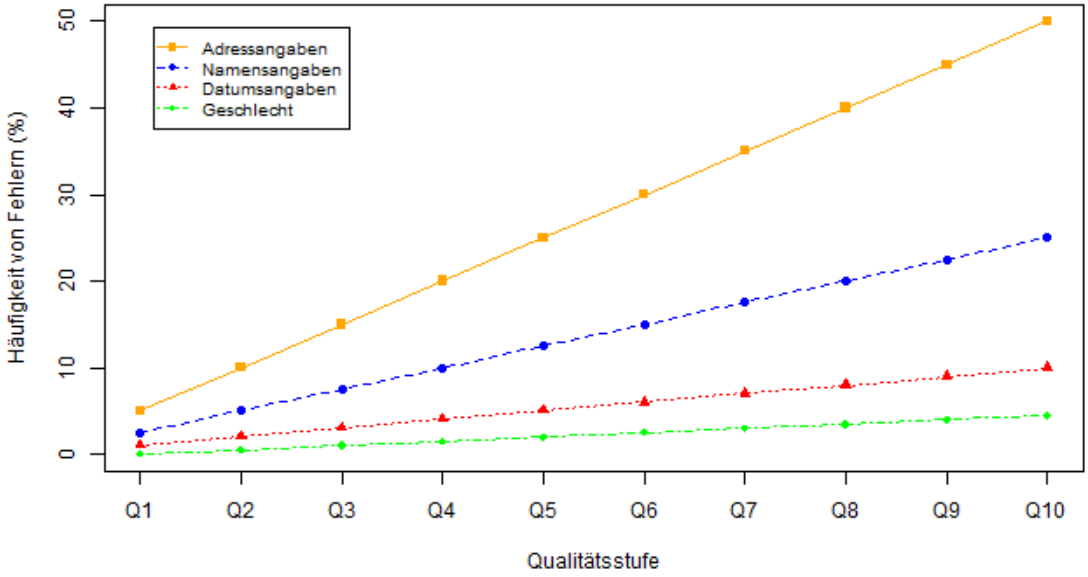


Abbildung 15: Mögliche Fehlerhäufigkeiten in Testsets abhängig von Qualitätsstufe und Attributsgruppe.

Grundsätzlich gibt es verschiedene Vorkommen von Fehlern, die in den verschiedenen Attributsgruppen verschieden häufig vorkommen. Diese wurden nach eigenem Ermessen wie folgend spezifiziert:

- Deformationsfehler: Fehler, die eine Ausprägung in eine nicht valide Ausprägung umwandeln.
- Transformationsfehler: Fehler, die eine Ausprägung in eine andere valide Ausprägung umwandeln.
- Fehlender Wert: Die Entität besitzt für dieses Element keine Ausprägung.

Fehlende Werte sind besonders häufig bei den Adressangaben, aber generell in jedem Feld beobachtbar. Abgleiche mit fehlenden Werte werden im Record-Linkage neutral gewichtet. Dies bedeutet, dass es beim Auftreten von fehlenden Werten in echten Übereinstimmungen schwierig fallen kann, diese wegen niedrigerem Gewicht als echte Links zu klassifizieren.

Bei Fehlern im Feld Geschlecht, bzw. in Datumsangaben handelt es sich meist um Transformationsfehler. Dies heißt, eine Attributsausprägung wird in eine tatsächlich vorkommende andere Ausprägung umgewandelt. Auch im Namen und den Adressfeldern dürfte die Mehrzahl der Fehler auf Transformationsfehler zurückzuführen sein. Als Beispiel sei

der Name „Meyer“ zu nennen. Geläufige Fehler dürften hierzu gleichklingende Namensvarianten sein wie beispielsweise „Meier“. Doch nicht nur phonetisch gleichklingende Namen bereiten hier Probleme. Auch Namensvarianten wie „Christa“ bzw. „Christel“ führen zu Transformationsfehlern. Weitere Transformationsfehler treten beispielsweise durch Namensänderung (z.B. Eheschließung) oder Adressänderungen auf. Dies kann zu einer positiven Gewichtung von Links führen, bei denen es sich eigentlich nicht um echte Übereinstimmungen handelt, und einer gleichzeitigen Verringerung des Gesamtgewichtes der tatsächlich übereinstimmenden Patienten. Transformationsfehler erhöhen demnach die Verwechslungsgefahr mit anderen Individuen.

Deformationsfehler, die in komplett neuen Varianten resultieren, dürften eher seltener sein. Diese treten nur dann auf, wenn eine Ausprägung etwa durch das zufällige Hinzufügen oder Weglassen eines Buchstabens so stark verändert wird, dass ein neuer, in der Werteverteilung bisher noch nicht aufgetretener Ausprägungswert geschaffen wird. Das Weglassen des Buchstabens „r“ im Namen „Christoph“ würde so in der Ausprägung „Chistoph“ resultieren. Dies wäre eine Ausprägung, die wohl in dieser Form nicht in normaler Namensverteilung vorkommen würde. Deformationsfehler führen demnach beim Durchführen des Kontrollnummerabgleichs, ähnlich wie bei fehlenden Werten, im Normalfall zu einer generell schwächeren Gewichtung.

Zu den Häufigkeiten der vorkommenden Fehler in medizinischen Daten konnten keine Angaben gefunden werden. Die Fehlerhäufigkeiten wurden aus diesem Grund heuristisch, also basierend auf eigenen Erfahrungen, geschätzt (siehe *Tabelle 11* sowie *Abbildung 16*).

Tabelle 11: Häufigkeit von Fehlerarten in Abhängigkeit der gegebenen Attributsgruppe.

Attributsgruppe	Transformation	Deformation	Fehlender Wert
Namensattribute	70%	20%	10%
Datumsangaben	80%	0%	20%
Adressangaben	40%	30%	30%
Geschlecht	70%	5%	25%

Entsprechend der gegebenen Häufigkeiten wurden nun Fehler in die Kontrollnummern der Patienten innerhalb des Überlappungsbereiches des kleineren Teilsets eingefügt. (siehe *Abbildung 14d*). Hierbei wurde für jede Attributsausprägung ein zufälliger Fließkommawert zwischen 0 und 100 generiert und mit den gegebenen Fehlerraten abgeglichen. Lag der Wert unter dem gegebenen Schwellwert wurde ein Fehler nach nachfolgendem Schema erzeugt.

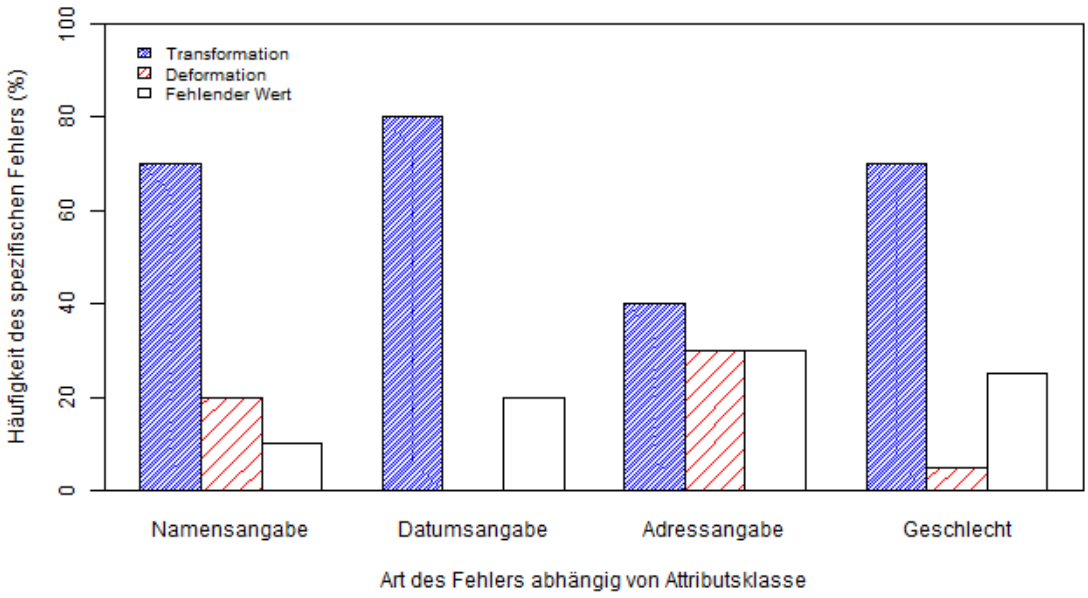


Abbildung 16: Häufigkeit der Fehlerart in Abhängigkeit der gegebenen Attributsgruppe.

Bei Deformationsfehlern wurde ein zufälliges Symbol in den Hash-Werten durch ein nicht im Hexadezimalcode vorkommendes Zeichen ersetzt. Hierdurch entstanden neue deformierte Werte, die in dieser Form außer bei Auftreten des exakt selben Fehlers bisher nicht in der Wertemenge enthalten waren.

Bei Auftreten von Transformationsfehlern wurde die alte Ausprägung durch eine neue aus der Gesamtwertemenge der Klinikumsdaten stammende Ausprägung ersetzt.

Bei fehlenden Werten wurde der alte Hash-Wert durch einen leeren String ersetzt.

Übertragung der Matches in das Teilset B

Der mit Fehlern versehene Überlappungsbereich konnte nun an die bereits bestehende Liste an Einträgen in Teilset B angehängt werden (siehe *Abbildung 14 e*).

2.3.4. Auswertung der Testsets

In den nachfolgenden Analysen (siehe *Kapitel 2.6*) galt es unter anderem, die Güte verschiedener binärer Klassifikatoren auf den 400 gegebenen Testsets zu prüfen. Hierbei war nicht nur der Vergleich der Klassifikatoren untereinander interessant, sondern auch die Information, wie nahe sich diese Klassifikatoren mit ihrer Vorhersage qualitativ an die auf dem jeweiligen Testset bestmögliche Güte annähern konnten. Es galt also, initial zu jedem Testset

die bestmögliche Güte zu bestimmen. Definiert wurde diese in dieser Arbeit als der vom jeweiligen Testset abhängige maximale F-Measure-Wert, der durch eine binäre Klassifikation auf dem ausgewählten Testset erzielt werden kann. Die nachfolgenden Unterkapitel erläutern, wie bei der Bestimmung der testsetspezifischen, maximalen F-Measure-Werte vorgegangen wurde.

Record-Linkage auf den Testsets

Zu jedem der 400 Testsets wurde mithilfe des unter *Kapitel 2.1.2* beschriebenen Systems ein Record-Linkage durchgeführt. Somit wurden 400 testsetabhängige Gewichtsdateien erzeugt, auf denen weiterführend der jeweils bestmögliche F-Measure-Wert berechnet werden konnte. Die Automatisierung des Record-Linkage auf den 400 gegebenen Testsets wurde mithilfe des Programmes *CreateTestSetsWeights<8>* realisiert.

Bestimmung des optimalen F-Measure-Wertes

Zu den testsetspezifischen Gewichtsdateien wurde der jeweils höchstmögliche F-Measure-Wert berechnet. Der Algorithmus hierzu war trivial. Zu einem Schrankenwert, der die Gewichtsdatei in echte und falsche Links unterteilte, ließen sich jeweils anhand der bekannten ID Übereinstimmungen zwischen Teilset A und Teilset B die TP, FP sowie FN berechnen. Aus diesen Bemessungen ließ sich zum gegebenen Schrankenwert jeweils der F-Measure-Wert berechnen. Angefangen beim niedrigsten in der jeweiligen Gewichtsdatei anfangenden Gewichtswert wurde diese Schranke inkrementell um einen Wert von jeweils 0,1 in Richtung höherer Gewichte verschoben. An jeder Position erfolgte eine Berechnung des F-Measure-Werts . Der Maximalwert wurde gespeichert und in eine Datei geschrieben.

Das Inkrement von 0,1 hätte grundsätzlich auch kleiner gewählt werden können, um eine noch genauere Messung zu gewährleisten, resultierte aber in einer dem Faktor entsprechenden linearen Laufzeit-Erhöhung der Prozedur. Für diese Arbeit erschien eine Approximation auf eine Nachkommastelle jedoch ausreichend. Somit muss dem Leser an dieser Stelle klar sein, dass es theoretisch auch höhere Maximalwerte für den F-Measure-Wert gäbe, was jedoch nur dann der Fall wäre, wenn mehrere Links ein unterschiedliches Gewicht innerhalb eines Gewichtsintervalles von 0,1 besäßen.

Die konkrete Implementierung hierzu fand sich im Programm *FMeasure<9>*, mittels dessen die automatisierte Berechnung des F-Measure-Wertes auf allen 400 gegebenen Gewichtsdateien durchgeführt wurde. Die Ergebnisse wurden dabei gesammelt in eine Datei übertragen. Weitere Optimierungsverfahren hierzu wären denkbar.

Bestimmung der optimalen Schranke

In der Praxis findet sich oft ein optimaler F-Measure-Wert, der sich nicht nur auf eine Gewichtsposition beschränkt, sondern ein größeres Gewichtsintervall abdecken kann. *Abbildung 3a* verdeutlicht diesen Fakt. Der optimale F-Measure-Wert ist hierbei zwischen den beiden Erhebungen zu erwarten, eine Klassifikationsschranke würde also unabhängig von der Position innerhalb des Intervalls zwischen den beiden Erhebungen im selben F-Measure resultieren. Für die überwachte Klassifizierung, die in den nachfolgenden Kapiteln näher vorgestellt wird, musste jedoch auf Trainingsdaten ein exakter Schrankenwert zum gegebenen maximalen F-Measure-Wert bestimmt werden, der später auf den Testdaten als Klassifikator verwendet werden konnte. Die Festlegung dieses Wertes wurde wie nachfolgend gehandhabt:

1. Gibt es ein Gewichtsintervall, über das sich der maximale F-Measure-Wert streckt, so wird als optimaler Schrankenwert der Mittelwert dieses Intervalls spezifiziert.
2. Gibt es mehrere Intervalle dieser Art, so wird das breiteste Intervall zur Ermittlung der Schranke, gewählt und Regel 1 wird auf dieses Intervall angewandt.

Graphische Auswertung in Bezug auf die Parametrisierung

Die individuellen Testsets wurden anhand der Kombination verschiedener Konstruktionsparameter erzeugt. Interessant war es hierbei, ob und inwiefern die verschiedenen Konstruktionsparameter einen Einfluss auf die bestmögliche Klassifikationsqualität besaßen.

Hierzu wurden die zu den 400 Testsets ermittelten maximalen F-Measure-Werte jeweils entsprechend der möglichen Ausprägungen der genannten Parameter gruppiert und der durchschnittliche F-Measure-Wert innerhalb dieser Gruppen abhängig von der Ausprägung des Parameters grafisch dargestellt. Die Ergebnisse hierzu finden sich unter *Kapitel 3.1.*

2.4. **Identifikation von potentiell einflussreichen Parametern auf die Erzeugung von Trainingssets**

Wie Sariyar [83,84] beschreibt, können gerade überwachte Klassifizierungssystemen im Bereich des Record-Linkage zu einer hohen Datenabgleichsgüte beitragen. Als offenes Problem nennen die Autoren jedoch Unklarheit über die genaue parametrische Beschaffenheit, wie beispielsweise die Bestimmung der Größe der zugrunde liegenden Trainingssets.

Um die Parametrisierung der Trainingsset in Bezug auf überwachte Klassifizierung zu normieren, und um hierbei ein mögliches Optimum zu ermitteln, wurde zu dieser Arbeit folgende Hypothese aufgestellt:

Je ähnlicher ein Trainingsset dem zu prüfenden Testset ist, umso ähnlicher sind auch deren optimale Klassifikatoren.

Die Interpretation hierzu lautete: Konstruktionsparameter, wie beispielsweise die Größe der Teilsets, die zur Konstruktion von Trainingssets verwendet wurden, sollten denen der Ausgangsdaten möglichst entsprechen.

Diese Hypothese mag nachvollziehbar klingen, wie Han et Al. [41] in diesem Zusammenhang jedoch kommentieren, besteht bei solch einer Hypothese immer die Gefahr eines Overfittings, also einer Überanpassung der Trainingsdaten an die Ausgangsdaten. Zudem durften die Trainingsdaten offensichtlich mit den Originaldaten nicht komplett übereinstimmen. Es musste also ein Kompromiss zwischen Anpassung und Differenzierung gefunden werden. Diese Differenzierung war in Bezug auf überwachte Klassifizierung jedoch bereits intrinsisch gegeben, wenn man bedenkt, dass der echte Überlappungsbereich nicht bekannt war. Die Differenzierung sollte also in der Erzeugung eines neuen Überlappungsbereiches, der für eine überwachte Klassifikation notwendig war, erfolgen. In Bezug auf die medizinische Domäne musste es also Patienteneinträge in Trainingsset A geben, die sich auch in Trainingsset B wieder fanden, und die Beziehung dieser Einträge musste über eine identische ID gekennzeichnet werden. Entsprechend der Größe des definierten Überlappungsbereiches in den Trainingsdaten mussten also als Mindestvoraussetzung mit zusätzlicher ID gekennzeichnete Einträge aus Trainingsset A nach Trainingsset B kopiert werden. Versuchte man hierbei die Trainingsdaten möglichst stark an die Testdaten anzupassen, so hätte man Trainingsset A (zuzüglich neuer ID) als direkte Kopie von Testset A erzeugen können. Trainingsset B hingegen hätte man als eine Kopie von Testset B erzeugen können, abzüglich einer Anzahl zufälliger Patienten, die der Größe des neuen Überlappungsbereich entsprochen hätte. Das Trainingsset B hätte man dann noch mit einer Liste zufälliger Patientenkopien aus Trainingsset A aufgefüllt.

Hierbei stellten sich nun einige Fragen. Käme dieses Vorgehen einer möglichst starken Anpassung der Trainingsdaten an die Testdaten, das aus der genannten Hypothese abgeleitet wurde, der Datenabgleichsqualität tatsächlich zugute? Gleich bleibende Teilsetgrößen waren bereits Teil des zuvor genannten Vorgehens, doch wie war es mit der Größe des Überlappungsbereiches? Hatte eine Anpassung des Überlappungsbereiches in den

Trainingsdaten auf die Größe des Überlappungsbereiches in den Testdaten ebenfalls eine positive Auswirkung? War es notwendig, die Fehlerraten im Überlappungsbereich der Trainingsdaten möglichst an die der Testdaten anzupassen? War es überhaupt sinnvoll, sich direkt an den Originaldaten zu bedienen, also die Werteverteilung der Trainingsdaten an denen der Testdaten möglichst zu orientieren?

Eine Überprüfung, ob die genannte Hypothese korrekt war und wie sie methodisch interpretiert werden konnte, war Teilaufgabe dieser Arbeit.

Zu den genannten Parametern, Größe der Teilsets, Größe des Überlappungsbereiches, Fehlerraten im Überlappungsbereich, sowie die Werteverteilung sollten deswegen nachfolgend Untersuchungen vorgenommen werden, um zu prüfen, ob sich eine Anpassung dieser Werte an die Originaldaten positiv auf die Klassifikation eines probabilistischen Record-Linkage-Systems auswirkten oder nicht. Sollte dies für alle der genannten Parameter der Fall sein, wäre die zuvor aufgestellte Hypothese bestätigt.

Nicht geprüft wurden die Anpassung der Domäne bzw. der Datenstruktur an die Trainingsdaten. Es erschien offensichtlich, dass beispielsweise eine Erhöhung der Attributsanzahl in den Trainingsdaten zu einer durchschnittlich höheren Gewichtung von Datenvergleichen führen würde, was in Bezug auf eine möglichst übereinstimmende Klassifikation zwischen Trainings- und Testdaten kontraproduktiv gewesen wäre. Aus diesem Grund wurden in den folgenden Analysen stets Trainingssets mit übereinstimmender Datenstruktur aus derselben Domäne (Patientendaten) verwendet.

2.5. Überprüfung des Einflusses von Konstruktionsparametern auf die Qualität der Klassifikation

2.5.1. Zielsetzung der Parameterprüfung

In den nachfolgenden Kapiteln sollte geprüft werden, ob eine Anpassung der unter *Kapitel 2.4* identifizierten, zur Konstruktion der Trainingssets genutzten Parameter an die Ausgangsdaten tatsächlich zu einer verbesserten überwachten Klassifizierung führte. Sollte sich zeigen, dass die Anpassung aller identifizierten Parameter einen positiven Einfluss auf die Klassifizierung ausübte, wäre dies ein Indiz für die Hypothese aus *Kapitel 2.4*. Unabhängig davon sollte aber versucht werden, die Klassifikationsqualität durch eine Bestimmung passender

Parameterwerte zu maximieren und eine hierauf basierende Methodik zur überwachten Klassifikation bei probabilistischen Record-Linkage-Systemen zur Verfügung zu stellen.

Hierfür sollten zu jedem Testset als Template-Trainingsset bezeichnete Datensets erstellt werden. Diese sollten entsprechend der Hypothese aus *Kapitel* 2.4 mit möglichst hoher Ähnlichkeit zu den Orginal-Trainingssets erstellt werden. Bei der Konstruktion sollten also die Größe der Teilsets, die Größe des Überlappungsbereiches, Fehlerraten sowie die Verteilungswerte möglichst zwischen Template-Trainingsset und Testset übereinstimmen. Die genaue Konstruktion wird unter *Kapitel 2.5.2* näher erläutert.

Auf den Teilsets der Template-Trainingssets konnte anschließend ein Record-Linkage vollführt werden. Auf jeder der erzeugten Template-Gewichtsdateien konnte schließlich ein Klassifikator, der den F-Measure-Wert auf dem jeweiligen Template-Trainingsset maximiert, berechnet werden. Die hierbei erzeugten optimalen Schranken konnten wiederum als überwachte Klassifikatoren auf den zugrunde liegenden Testsets verwendet werden.

Zu diesem Zeitpunkt hätte sich also bereits ermitteln lassen, wie stark die überwachte Klassifikation, basierend auf der Template-Parametrisierung, von der bestmöglichen Klassifizierung auf dem zugrunde liegenden Testset (siehe *Kapitel 2.3*) abwich. Ferner galt es jedoch zu prüfen, ob es sich bei der Parametrisierung der genannten Template-Trainingssets wirklich um eine optimale Parametrisierung handelte oder ob es Varianten in der Parametrisierung gab, die zu noch besseren Ergebnissen führten. Aus diesem Grund sollten weitere Trainingsset-Varianten erzeugt werden, die jeweils in einem der Konstruktionsparameter von den Template-Trainingssets abwichen. Die Trainingsset-Varianten werden in den nachfolgenden Kapiteln näher erläutert. Zu diesen Varianten sollte entsprechend dem Klassifikationsvorgang bei der Template-Variante erst der jeweils optimale Schrankenwert (bemessen am F-Measure-Wert) auf den jeweiligen Trainingsset-Varianten bestimmt werden und dieser dann als Klassifikator auf das korrespondierende Testset angewendet werden. Erneut ließ sich hierbei zu jeder Variante die Performanz des vorhergesagten Klassifikators, also der F-Measure-Wert berechnen. Erzielten die auf den Trainingsset-Varianten basierenden Klassifikatoren auch nur zum Teil bessere Gütewerte als die Klassifikation auf den Template-Trainingssets, so wäre die ursprüngliche Hypothese widerlegt und die Parameter wären für ein finales Modell entsprechend der besser abschneidenden Variante anzupassen. Die Ergebnisse der beschriebenen Analyse finden sich unter *Kapitel 3.2*. *Abbildung 17* illustriert den eben genannten experimentellen Ansatz.

2.5.2. **Erstellen von Template-Trainingssets**

Zu jedem der 400 unter *Kapitel 2.3.* erstellten Testsets wurde ein der Hypothese möglichst entsprechendes Template-Trainingsset erzeugt. Dieses sollte mit dem Originaltestset jeweils in Größe der Teilsets, Größe des Überlappungsbereiches sowie in der Häufigkeit auftretender Fehler im Überlappungsbereich möglichst gut übereinstimmen. Die genauen Konstruktionsparameter wurden hierbei über den Dateinamen der Testdaten übergeben (siehe *Abbildung 13*). Weiterhin sollte sich die Verteilung der Werteausprägungen stark an den Originaldatei orientieren. Das genaue Vorgehen zur Erzeugung der Template-Trainingssets wird unter *Abbildung 18* bildlich dargestellt und weiterführend beschrieben. Zu Teilset A des Testdatensatzes wurde wie schon im Falle der Testseterzeugung eine identische Kopie erstellt (siehe *Abbildung 18a*). Jeder Eintrag in diesem neuen Trainingsset A wurde jedoch zusätzlich noch mit einer neuen ID eindeutig markiert. Zu Teilset B des zugrunde liegenden Testdatensatzes wurde ebenfalls eine identische Kopie erstellt (siehe *Abbildung 18b*). Allerdings wurden aus dem hierbei erstellten Trainingsset B eine zufällige Auswahl an Patienten entfernt. Die Anzahl entsprach dabei der Größe des Überlappungsbereiches. Aus Trainingsset A wurden nun zufällige Patienten entsprechend der Größe des originalen Überlappungsbereiches ausgewählt. Diese bildeten den neuen Überlappungsbereich (siehe *Abbildung 18c*). In den neuen Überlappungsbereich wurden entsprechend den Originaldaten Fehler eingefügt (siehe *Abbildung 18d*). Die genauen Fehlerhäufigkeiten wurden dabei über den Dateinamen der Testdaten übergeben. Der neu konstruierte, mit Fehlern versehene Überlappungsbereich, der unter Schritt d erzeugt wurde, wurde mit dem in Schritt b erzeugtem Datenset vereint und bildete das neue Trainingsset B (siehe *Abbildung 18e*). Die beiden konstruierten Teilsets bildeten nach vorhergehendem Schema ein auf ein Testset angepasstes Template-Trainingsset.

Die automatisierte Erzeugung der 400 auf den Testsets beruhenden Template-Trainingssets wurde mithilfe des Programmes *CreateTemplateTrainingsset*<10> realisiert. Nachfolgend wurden die jeweils einzelnen Teilsets der 400 Template-Trainingssets per Record-Linkage (*CreateTrainingSetsWeights<11>*) abgeglichen, was in 400 Gewichtsdateien resultierte.

Zu jeder dieser Template-Gewichtsdateien wurde schließlich mit Hilfe der Programme *MassFMeasures<12>* analog zu *Kapitel 2.3,* erst ein maximaler F-Measure-Wert und anschließend jeweils ein hierauf basierender optimaler Template-Schrankenwert bestimmt. Dieser vorhergesagte Template-Schrankenwert wurde nun wiederum als Klassifikator, also als Schrankenwert für das jeweilige Testset, wieder verwendet und dessen Qualitätsgüte auf den

Testdaten (F-Measure) dokumentiert. Der Name des hierzu verwendeten Programmes lautet *FitBorderToTestset<13>.*

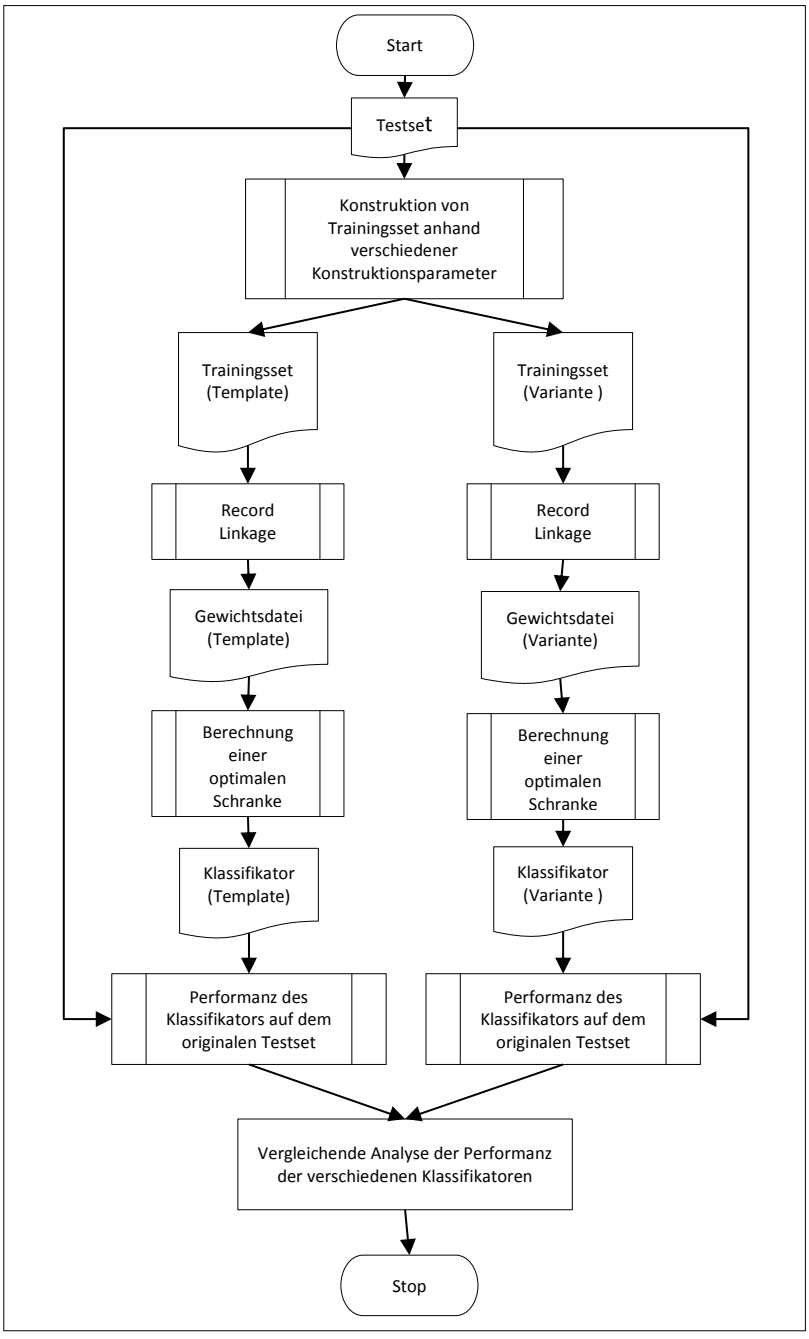


Abbildung 17: Schematischer Ablauf des Performanzvergleiches zwischen Klassifikatoren eines Template-Trainingssets und einer Trainingsset-Variante.

Die Template-Trainingssets setzten im Grunde genommen das exakte Wissen über die Parametrisierung (hier über den Dateinamen gegeben) voraus. Im Echteinsatz wären diese Parameter jedoch nicht ohne weiteres exakt ermittelbar. Eigentlich würde es sich um das sogenannte Henne-Ei-Problem handeln [98]. Um den Überlappungsbereich zu bestimmen, bzw. durch einen Klassifikator abzugrenzen, hätte die Größe des Überlappungsbereiches im Vorfeld bekannt sein müssen, was zwar auf Testdaten gegeben war, auf Realdaten jedoch nicht. Als Abhilfe hätte es zu diesem Beispiel theoretische Möglichkeiten gegeben, die Größe des Überlappungsbereiches grob abzuschätzen [71]. Es wären allerdings weitere Untersuchungen über die Qualität dieser Abschätzungen und Auswirkungen auf eine Klassifizierung, die auf Template-Trainingssets beruht, erforderlich gewesen.

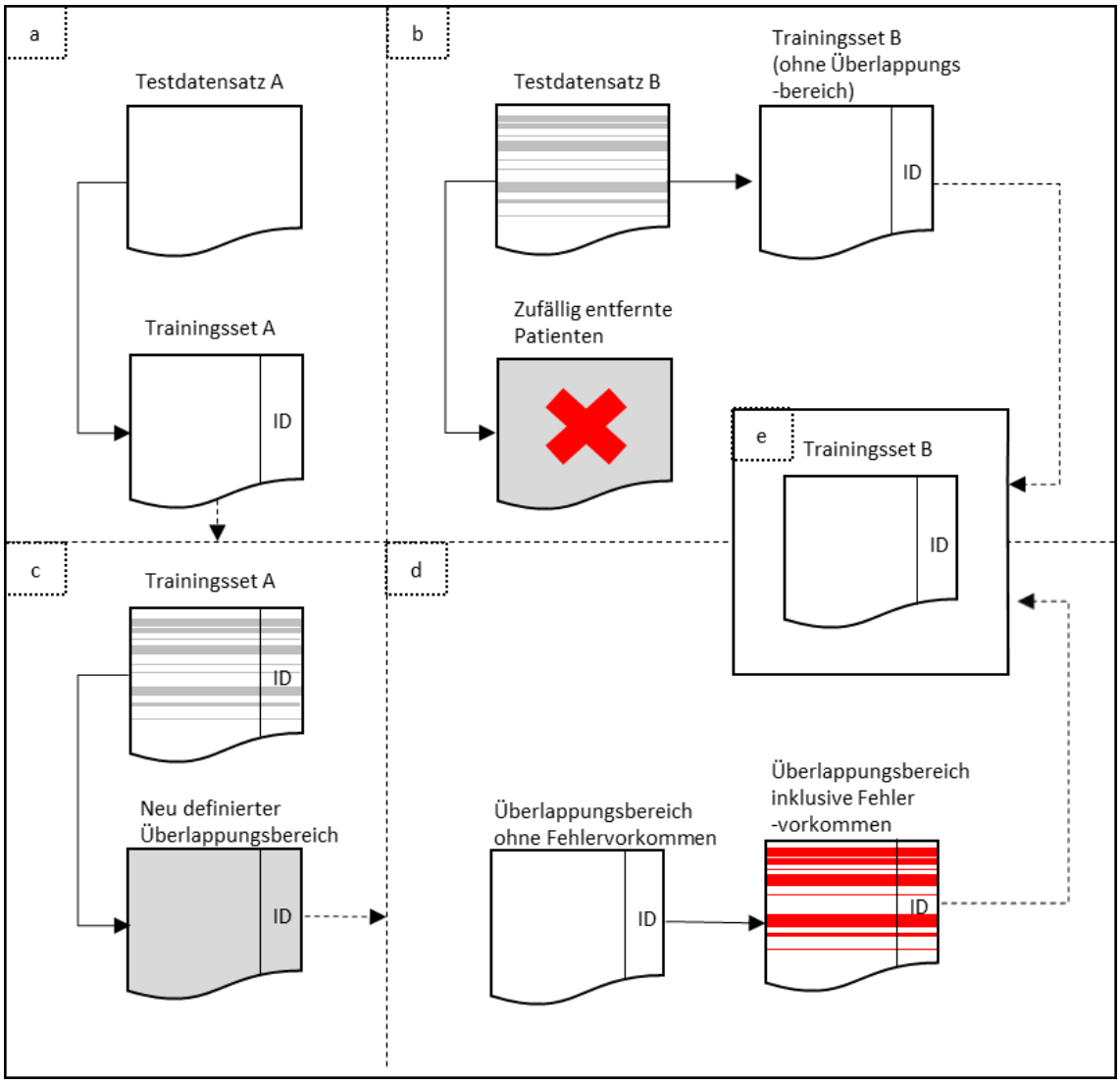


Abbildung 18: Erzeugung von auf spezifische Testsets angepasste Template-Trainingssets.

2.5.3. Variation der Größe

Zu jedem Testset wurden zusätzlich jeweils zwei Trainingsset-Varianten erstellt, die in der Größe von der Template-Parametrisierung abwichen. Hierbei galt es zu prüfen, ob die Klassifitkationsqualität abwich, wenn nicht die exakten Größen der Testdaten zur Konstruktion der Trainingsdaten verwendet wurden.

Bei der ersten Variante wurde der Größenparameter für jeweils beide Teilsets der Trainingsset-Variante auf 100 festgelegt. In den meisten Fällen resultierte dies in einer Reduktion der Größe der Teilsets im Vergleich zu den Testdaten. Bei Teilset A dieser Variante handelte es sich also nicht um eine direkte Kopie von Testset A sondern um eine zufällige Ziehung von exakt 100 Patienten. Teilset B dieser Variante wurde analog entsprechend *Abbildung 18* mit 100 zufälligen Patienten (abzüglich der Größe des Überlappungsbereiches) aus Testset B befüllt. Der Überlappungsbereich wurde analog zum Template-Trainingsset mit zufälligen Einträgen aus Trainingsset A befüllt und entsprechend dem Testset mit Fehlern versehen. Trainingssets dieser Variante wurden mithilfe des Programmes *CreateSizeVariant1Trainingsset<14>* erzeugt.

Bei der zweiten Größenvariante wurde die Größe der Teilsets wie bei der ersten Variante nicht auf einen konstanten Wert festgelegt, sondern die Variante orientierte sich an den ursprünglichen Größenverhältnissen im Testset. Die Größe wurde hierbei jeweils halbiert, das Größenverhältnis blieb also erhalten. Trainingssets dieser Variante wurden mithilfe des Programmes *CreateSizeVariant2Trainingsset<15>* erzeugt.

2.5.4. Variation der Fehlerrate

Zur Prüfung, ob die Erhaltung der exakten Fehlerrate bei der Konstruktion der Trainingssets eine Rolle spielte, wurde eine Trainingsset-Variante konstruiert, bei der komplett auf Fehler im Überlappungsbereich verzichtet wurde. Trainingssets dieser Variante wurden mithilfe des Programmes *CreateErrorVariantTrainingsset<16>* erzeugt.

2.5.5. Variation der Überlappung

Um zu prüfen, inwiefern die Klassifikationsqualität bei Variation der Größe des Überlappungsbereiches von der Klassifikation bei Verwendung des Template-Trainingssets abwich, wurde in dieser Variante für die Größe des Überlappungsbereiches nicht der

Originalwert der Testdaten sondern ein fester Standardwert verwendet. Hierbei wurden zu allen der 400 Testdatensätze jeweils drei Varianten entworfen mit festen Standardwerten von jeweils 3%, 30% sowie 90% in Bezug auf die Anzahl von Patientendaten innerhalb des Überlappungsbereiches. Die Prozentzahlen bezogen sich, wie bereits unter *Kapitel 2.3.2* beschrieben, auf das jeweils kleinere Teilset. Trainingssets dieser Varianten wurden mithilfe der Programme *CreateOverlapVariant1Trainingsset<17>, CreateOverlapVariant2Trainingsset<18>* sowie *CreateOverlapVariant1Trainingsset<19>* erzeugt.

2.5.6. **Variation der Verteilung**

Letztendlich wurde geprüft, ob es Sinn macht, die Verteilung der Werte in Testsets bestmöglich zu erhalten, oder ob die Werteverteilung eine eher vernachlässigbare Rolle bei der Klassifizierung spielte. Rekapitulierend: Bei den Template-Trainingssets war das Trainingsset A jeweils die direkte Kopie des zugrunde liegenden Testsets A. Die Verteilung der Werte stimmte hier also exakt überein. Trainingsset B orientierte sich ebenfalls an den Testdaten, variierte aber im Überlappungsbereich, bei dem es sich um direkte Kopien aus Trainingsset A handelte. Es sollte sich also auch in Trainingsset B um eine zumindest ähnliche Verteilung wie in Teilset B handeln.

Bei der im Folgenden erläuterten, neuen Verteilungsvariante jedoch wurden die Trainingssets nicht wie bisher üblich mit den direkten Kopien aus den zugrunde liegenden Testsets befüllt. Anstelle der Template-Prozedur wurden die Trainingssets dieser Variante mit künstlich assemblierten Patienten belegt.

Künstlich assemblierte Patienten bezogen ihre Ausprägungen (Attributswerte) direkt aus der Wertemenge des kompletten Basisdatenbestandes des Klinikums. Frequenzen und Häufigkeiten spielten hierbei keine Rolle, da die Chance, eine spezifische Ausprägung zu erhalten, gleich verteilt war. Anstelle eines Datensatzes von spezifischen Verteilungswerten bot diese Trainingsset-Variante also Klassifizierung basierend auf gleich verteilten Werten. Trainingssets dieser Variante wurden mithilfe des Programmes *CreateDistributionVariant1Trainingsset<20>* erzeugt.

2.5.7. Performanzvergleich der Klassifikatoren der Trainingsset-Varianten

Die am maximalen F-Measure-Wert kalibrierten Klassifikatoren des Template-Trainingssets sowie die sieben zuvor beschrieben Trainingsset-Varianten wurden entsprechend *Abbildung 18* auf die Testdaten angewandt und deren Klassifikationsgüte verglichen. Um Zufallsergebnisse auszuschließen und um die Interpretation der Ergebnisse zu erleichtern, wurden hierbei insgesamt drei komplette Sets an Trainingsvarianten bzw. Template-Trainingssets erzeugt. Das hierfür notwendige Hauptprogramm lautet *AutomateTrainingssetProduction<21>.* Insgesamt wurden also 9600 (siehe *Formel 16*) Trainingssets erzeugt und ausgewertet.

$$|Trainingssets| = 400 \times 8 \times 3 = 9600. \qquad (16)$$

Die Ergebnisse hierzu werden unter *Kapitel 3.2* näher beschrieben.

2.6. Vergleich von unüberwachter Klassifizierung mit anderen Klassifikationstechniken

2.6.1. Zielsetzung des Klassifikatorenabgleichs

Basierend auf den Ergebnissen aus *Kapitel 3.2* sollten die Parameter des Template-Trainingssets optimiert werden. Diese optimierte Variante der überwachten Klassifizierung galt es mit anderen zum Teil etablierten Klassifikationsmethoden auf den 400 erzeugten Testsets zu prüfen und die Performanz für einen möglichen Realeinsatz zu bewerten. Von Hauptinteresse war der Vergleich zu unüberwachten Systemen, die in der Praxis aufgrund der Unabhängigkeit von Trainingsdaten in der Regel den Vorzug bekommen. Hierbei wurde zum einen eine aus dem maschinellen Lernen bekannte Clustering-Methode, das Single-Linkage-Clustering [77], das es ermöglichen soll, Links korrekt zu zwei Clustern (echte Links/falsche Links) zuzuordnen, angewandt. Es war zu erwarten, dass diese Methode, die nicht unbedingt für das Record-Linkage konzipiert wurde, im direkten Vergleich eher schlecht abschneidet. Zum anderen wurde eine von Peter Christen vorgestellte Methode, die 2-Step-Seeded-K-Nearest-Neighbour-Klassifikation [71], in zwei Varianten mit den anderen Methoden abgeglichen. Zur Vereinfachung wurde die Methodik nachfolgend als SNN bezeichnet.

Letztere Methode wurde bereits mit anderen unüberwachten Klassifikationsmethoden verglichen und konnte hierbei Verbesserungen bei der Zuordnungsqualität im Bereich des Record-Linkage erzielen. Beispielsweise übertrifft die genannte Methode den Hybrid-TAILOR Ansatz, von dem wiederum gezeigt wurde, dass dieser andere aus dem maschinellen Lernen bekannte Klassifikationsmethoden, was die finale Abgleichsqualität angeht, übertrifft [93].

Final wurden die 400 Testdatensätze manuell, anhand der Histogramme, wie es in der Praxis oft üblich ist, durch den Autor dieser Arbeit klassifiziert. Bei letzterem Vorgehen handelte es sich um einen stark subjektiven Ansatz. Dennoch erschien es interessant, zumindest grob abzuschätzen, inwiefern die manuelle Schrankensetzung mit anderen Methoden mithalten konnte und ob die Anwendung automatisierter Methoden im Realeinsatz überhaupt gerechtfertigt war. In den nachfolgenden Kapiteln werden die verschiedenen Methoden genauer spezifiziert.

2.6.2. Überwachte Klassifizierung der Testdaten

Zu jedem der 400 Testsets wurde entsprechend den Erkenntnissen aus *Kapitel 3.2* jeweils ein parameter-optimiertes Trainingsset erzeugt. Dieser Vorgang wurde dreimal wiederholt. Der Grund hierfür war, dass somit zu jedem Testset mehrere auf überwachter Klassifizierung basierende Klassifikatoren zur Verfügung standen. Bei der Wahl eines Mittelwertes dieser Klassifikatoren kann also der maximal mögliche Fehler minimiert werden.

Konkret wurde die parameter-optimierte Trainigsset-Erzeugung im Programm *CreateFinalTrainingsset<22>* implementiert. Zu jedem Trainingsset wurde analog zu den vorhergehenden Analysen eine Schranke basierend auf dem optimalen F-Measure-Wert ermittelt. Diese Schranken wurden jeweils in das entsprechende Testset eingepasst, der F-Measure-Wert an dieser Position berechnet und für die weiteren vergleichenden Untersuchungen in einer Datei festgehalten.

2.6.3. Unüberwachte Klassifizierung der Testdaten

Single-Linkage-Clustering

Die Auswahl einer Clustering-Methode sollte zeigen, ob es möglich war, gute Klassifizierungen anhand nicht auf das Record-Linkage speziell angepasster und leicht zu implementierender Klassifizierungsverfahren zu erhalten. Für den Praxisgebrauch wäre dies von Vorteil, da kompliziertere Algorithmen wie beispielsweise SNN-Klassifikation für die meisten Projekte nur

mit entsprechend geschultem IT-Personal umsetzbar wären. Konkret wurde für die vergleichende Analyse eine vereinfachte Variante des Single-Linkage-Clustering (SLC) [77] implementiert. Grundsätzlich handelt es sich beim SLC um agglomeratives bzw. hierarchisches Clustering [99], wobei jeder einzelne Gewichtswert einer Gewichtsdatei als einzelner Basiscluster interpretiert wird und die Cluster solange vereint werden, bis nur noch zwei Cluster vorhanden sind. Diese Cluster enthalten schließlich die echten bzw. falschen Links. Zwei Cluster werden während des Vorganges immer dann vereint, wenn die Distanz zwischen den nächsten Werten der in Ihnen vorkommenden Gewichtswerte jeweils minimal im Vergleich zu anderen Clusterpaarungen ist. Generell besitzen Clustering-Methoden eine Laufzeit von $O(n^3)$, was auf den 400 Gewichtsdateien, mit bis zu 2.441.271 Gewichten, zeitlich nicht realisierbar gewesen wäre. Lediglich für das Single-Linkage-Clustering und das Complete-Linkage-Clustering existieren Methoden, deren Laufzeit sich durch clevere Implementierung, SLINK [77] bzw. CLINK[78], auf $O(n^2)$ drosseln lässt. Grundsätzlich war aber eine weitere Vereinfachung der SLC-Methodik innerhalb dieses Projektes möglich. Da Gewichtsdateien lediglich eindimensionale Daten beinhalten (Gewichtswerte), muss das SLC hierbei trivialerweise lediglich nach dem größten Abstand zwischen den Gewichtswerten suchen. Dies wurde über das Programm SingleLinkageNAIV<23> realisiert.

Seeded-Nearest-Neighbour-Klassifikation

In einer Arbeit von Christen [71] wird gezeigt, dass bekannte Algorithmen aus dem Bereich des maschinellen Lernens, konkret der k-Nearest-Neighbour-Algorithmus bzw. die Verwendung von Support-Vector-Maschinen, durch die Definition von Keimmengen, also auf den Originaldaten basierende Trainingsdaten mit offensichtlicher Klasse, sehr gut zur Klassifikation im Bereich des Record-Linkage genutzt werden können. Algorithmen dieser Art fasst man auch unter aktivem Lernen zusammen [79]. In einem ersten Schritt werden die Keimmengen anhand festzusetzender, Kriterien befüllt. Bei den Keimmengen handelt es sich, wie bereits erwähnt, um offensichtlich echte bzw. falsche Übereinstimmungen. Die Kriterien, ab wann ein Link einer Keimmenge zuzuordnen wäre, variieren von Fall zu Fall, es gibt hierzu also keine festen Vorgaben. Die in die Keimmengen übertragenen Links können dann in einem zweiten Schritt, in dem der eigentliche Algorithmus angewendet wird, als Trainingsdaten, die den Algorithmus trainieren, verwendet werden. In der genannten Arbeit von Peter Christen werden nur Vorschläge aber keine festen Richtlinien für die Auswahl der Keimmenge genannt. In dieser Dissertationsarbeit wurden deshalb zwei Varianten zur Auswahl der Keimmenge gewählt. Zum einem wurde eine in der Arbeit von Peter Christen vorgestellte Formel zur Abschätzung der Größe der positiven bzw. negativen Keimmenge verwendet (*siehe Formel 17*).

$$r = \frac{\min(|A|, |B|)}{|W| - \min(|A|, |B|)} \qquad (17)$$

|W| steht hierbei für die Anzahl der Gewichte, |A| für die Größe des Teilsets A sowie |B| für die Größe des Teilsets B. Bei der Rückgabe-Variablen r handelt es sich um das Größenverhältnis zwischen der positiven und der negativen Keimmenge. Die negative Keimmenge wurde in dieser Arbeit, vergleichbar zur Veröffentlichung von Peter Christen, auf 5% der Anzahl der Gewichte festgelegt (befüllt mit den niedrigsten 5% der Gewichte).

Zum anderen wurde eine Variante implementiert, bei der feste Grenzwerte verwendet werden. Links mit einem Gewicht über +45 wurden zur positiven Keimmenge, Links mit einem Gewicht unter -15 zur negativen Keimmenge hinzugefügt. Diese Grenzwerte basierten auf Erfahrungswerten zur Klassifikation der Daten und waren datensatzspezifisch. Es zeigte sich also bereits bei der Implementierung der Technologie, dass die Methode viele Unsicherheiten barg und eine passende Abschätzung der Keimmenge dringend voraussetzte. Das grundlegende Prinzip der Erzeugung der Keimmengen wird vereinfachend in *Abbildung 19* illustriert.

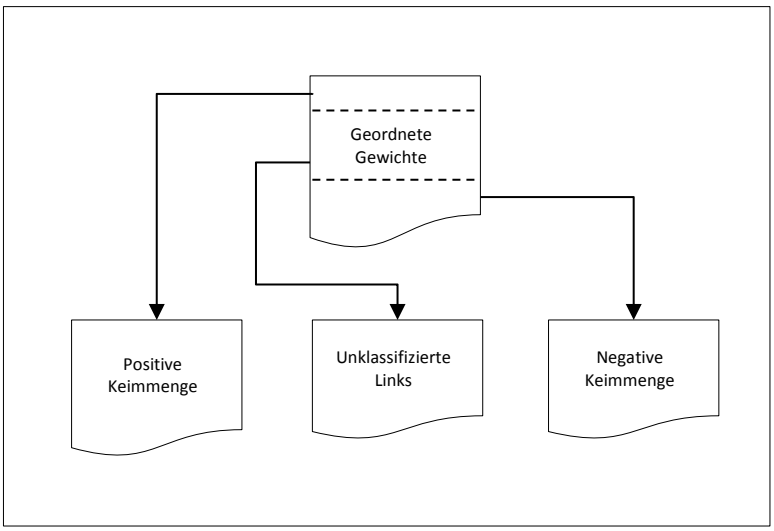


Abbildung 19: Aufteilung der Menge der Links in positive Keimmenge, negative Keimmenge sowie Menge der bisher unklassifizierten Links.

Nach Bestimmung der Keimmengen konnten die enthaltenen Links nun als Trainingsdaten für den eigentlichen Algorithmus genutzt werden. Für diese Arbeit wurde hierzu der K-Nearest-Neighbour-Ansatz implementiert. Der Algorithmus ließ sich wie folgend zusammenfassen. Ein bisher unklassifizierter Link wurde dann zu einer spezifischen Keimmenge hinzugefügt, wenn es sich bei diesem Link um den Link mit der niedrigsten Distanz zu k Links aus der vereinten

Keimmenge handelte, und sich mehr dieser nächsten benachbarten Links in der spezifischen positiven bzw. negativen Keimmenge befanden. Sobald alle unklassifizierten Links einer Keimmenge hinzugefügt wurden, war die Klassifikation abgeschlossen. Für diese Arbeit wurde der Wert k auf 3 festgelegt. Eine beispielhafte Illustration des Vorganges wird in *Abbildung 20* wiedergegeben. Hierbei ging es um die Klassifikation zweier bisher unklassifizierter Links. Zu den beiden Links wurde bestimmt, welcher der Links die minimale, aufsummierte Distanz zu den jeweils k nächsten Links aus der vereinten Keimmenge besaß (*Abbildung 20a*). In diesem Fall handelte es sich dabei um den Link mit niedrigerem Gewicht. Da seine nächsten drei Nachbarn der negativen Keimmenge angehörten, wurde der Link dieser Menge hinzugefügt (*Abbildung 20b*). Von den drei nächsten Nachbarn des letzten unklassifizierten Links befand sich die Mehrzahl in der positiven Keimmenge, wodurch der Link dieser Menge hinzugefügt wurde (*Abbildung 20c*). Es gab keine verbleibenden unklassifizierten Links. Die Klassifikation war somit abgeschlossen. Die sich in den Keimmengen unterscheidenden Algorithmen wurden in den Programmen *KNN_Seed1<24>* sowie *KNN_Seed2<25>* performant implementiert.

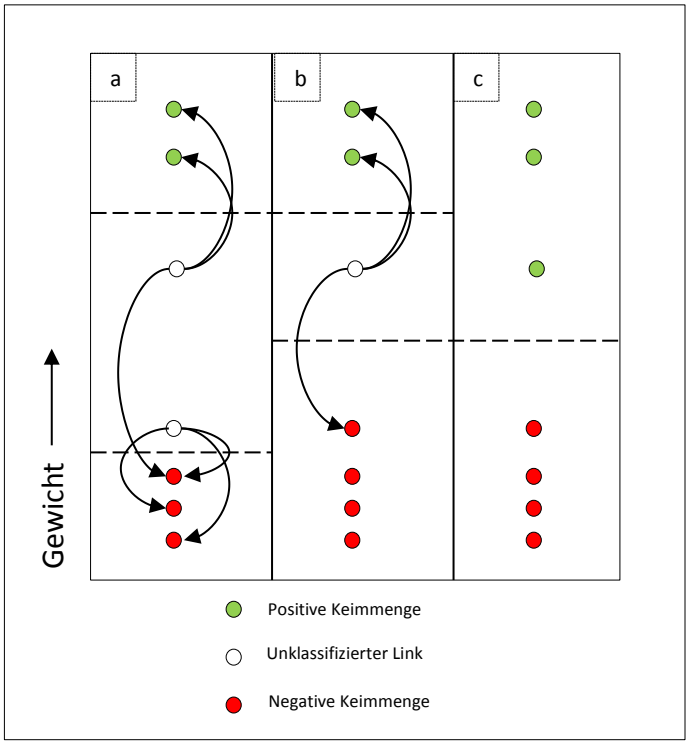


Abbildung 20: Beispielhafte Illustration des KNN-Algorithmus mit k=3.

Manuelle Klassifikation durch Auswertung der Testset-Histogramme

Für die manuelle Klassifikation anhand von Histogrammen wurden mithilfe des Programmes *CreateHistogramms<26>* zu jedem Testset Histogramm-Dateien erzeugt. Für die Klassifikation wurde hierbei eine zur DKFS analoge Darstellung gewählt (siehe *Abbildung 4*). Eine Schranke wurde dabei manuell gesetzt und der Wert in einer Datei hinterlegt. Eine Übersicht der Histogramme in kleinerem geordneten Format befindet sich in Anhang F. Ergebnisse

3. Ergebnisse

3.1. Testset-Erzeugung

Wie unter *Kapitel 2.3.* beschrieben, wurden anhand von Realdaten, die vom Klinikum Großhadern zur Verfügung gestellt wurden, 400 künstliche Testsets, die sich jeweils in mindestens einem der Konstruktionsparameter (Größe der Teilsets, Größe des Überlappungsbereiches, Beschaffenheit) unterschieden, erzeugt. Ein Testset bestand dabei jeweils aus einem Teilset A, sowie einem Teilset B. Diese Teilsets wurden jeweils per probabilistischem Record-Linkage abgeglichen. Zu den erzeugten Gewichtsdateien wurde jeweils der testsetspezifische maximale F-Measure–Wert berechnet. Um herauszufinden, inwieweit die Konstruktionsparameter im konkreten Fall die finale Klassifikationsqualität beeinflussten, wurden F-Measure-Werte anhand gleicher Ausprägung in den Konstruktionsparametern gruppiert, und der gemittelte F-Measure-Wert innerhalb dieser Gruppen bestimmt.

Abbildung 21 zeigt hierbei die gemittelten, maximalen F-Measure-Werte abhängig von den 10 innerhalb der Testsets auftreten Größenkombinationen der Teilsets. Jeder Messwert stellt hierbei den Durchschnittswert aus 40 Testsets mit der gegebenen Größenkombination dar.

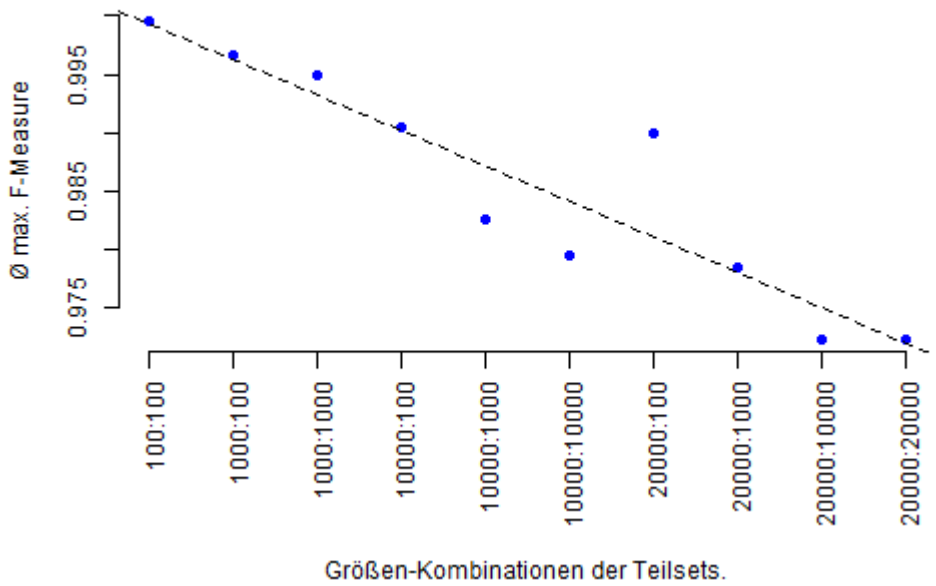


Abbildung 21: Gemittelter, maximaler F-Measure-Wert in Testsets mit spezifischer Größenkombination.

Wie sich zeigte war es schwierig, anhand der Grafik einen Trend, inwiefern die Größe der zugrunde liegenden Teilsets die Klassifikationsqualität beeinflusste, festzustellen. Es schien jedoch, dass das Matching auf Testsets, die kleine Teilsets enthalten, zu einer höheren, bestmöglichen Abgleichqualität führte. Der Befund deutete darauf hin, dass kleinere Trainingssets in weniger Vergleichen resultierten. Hierdurch ergaben sich eher lückenhafte, dünne Gewichtsdateien wie beispielsweise unter *Abbildung 3c* dargestellt. Größere Trainingssets neigten durch die Erhöhung der Vergleiche allein schon statistisch dazu, Übergangsbereiche zu verwischen (siehe *Abbildung 3b*). Auf dünnen Daten besaßen also optimale Klassifikatoren einen eher höheren maximalen F-Measure-Wert als auf dichteren Daten. Diese Aussage war natürlich auch stark abhängig von der gegebenen Datenqualität und dies sollte nicht implizieren, dass es generell leichter gewesen wäre, dünne Daten zu klassifizieren, da hier eine Fehlklassifikation (z.B. Auswahl der falschen „Lücke) wohl in einer größeren Abweichung vom echten Schrankenwert als auf dichten Daten resultiert hätte. Es war jedoch nicht auszuschließen, dass die Beobachtung auf eine andere Ursache, wie etwa die generelle Berechnung des F-Wertes zurückzuführen gewesen wäre. Zur besseren Darstellung wurden die Größenkombinationen auf zwei separate Achsen aufgebrochen (siehe *Abbildung 22*).

Abbildung 23 stellt den durchschnittlich höchstmöglichen F-Measure-Wert abhängig von der Größe des Überlappungsbereiches dar. Jeder Datenpunkt beinhaltet hierbei die Durchschnittswerte zu 100 verschiedenen Testsets. Es zeigte sich auf den gegebenen Daten, dass größere Überlappungsbereiche zwischen Teilsets in höheren, bestmöglichen F-Measure-Werten resultierten. Diese Beobachtung ließ sich mathematisch interpretieren. Der F-Measure-Wert stellte das harmonische Mittel der Sensitivität sowie des Positiv-Prädiktiven-Wertes dar. Bei Vergrößerung des Überlappungsbereiches erhöhte sich mit etwa gleich bleibendem Verhältnis die absolute Anzahl an True-Positives, sowie False-Negatives. Die Sensitivität sollte somit bei Variation des Überlappungsbereiches unbeeinflusst bleiben. Der Positiv-Prädiktive-Wert hingegen leitete sich aus der Anzahl der True-Positives sowie der False-Positives ab. Dieses Verhältnis veränderte sich bei Variation des Überlappungsbereiches jedoch, da die Anzahl der False-Positives bei Erhöhung des Überlappungsbereiches sich eher gleich bleibend, bzw. geringfügig absteigend verhalten sollte. Somit stieg der PPV tendenziell bei ansteigendem Überlappungsbereich, was wiederum in einer tendenziellen Erhöhung des F-Measure-Wertes resultieren würde.

Final wurden die durchschnittlich maximal erreichbaren F-Measure-Werte, abhängig von der zur Konstruktion verwendeten Qualitätsstufe, berechnet (siehe *Abbildung 24*). Jeder

Datenpunkt bestand hierbei jeweils aus den Ergebnissen von 40 in der Qualitätsstufe übereinstimmenden Testsets.

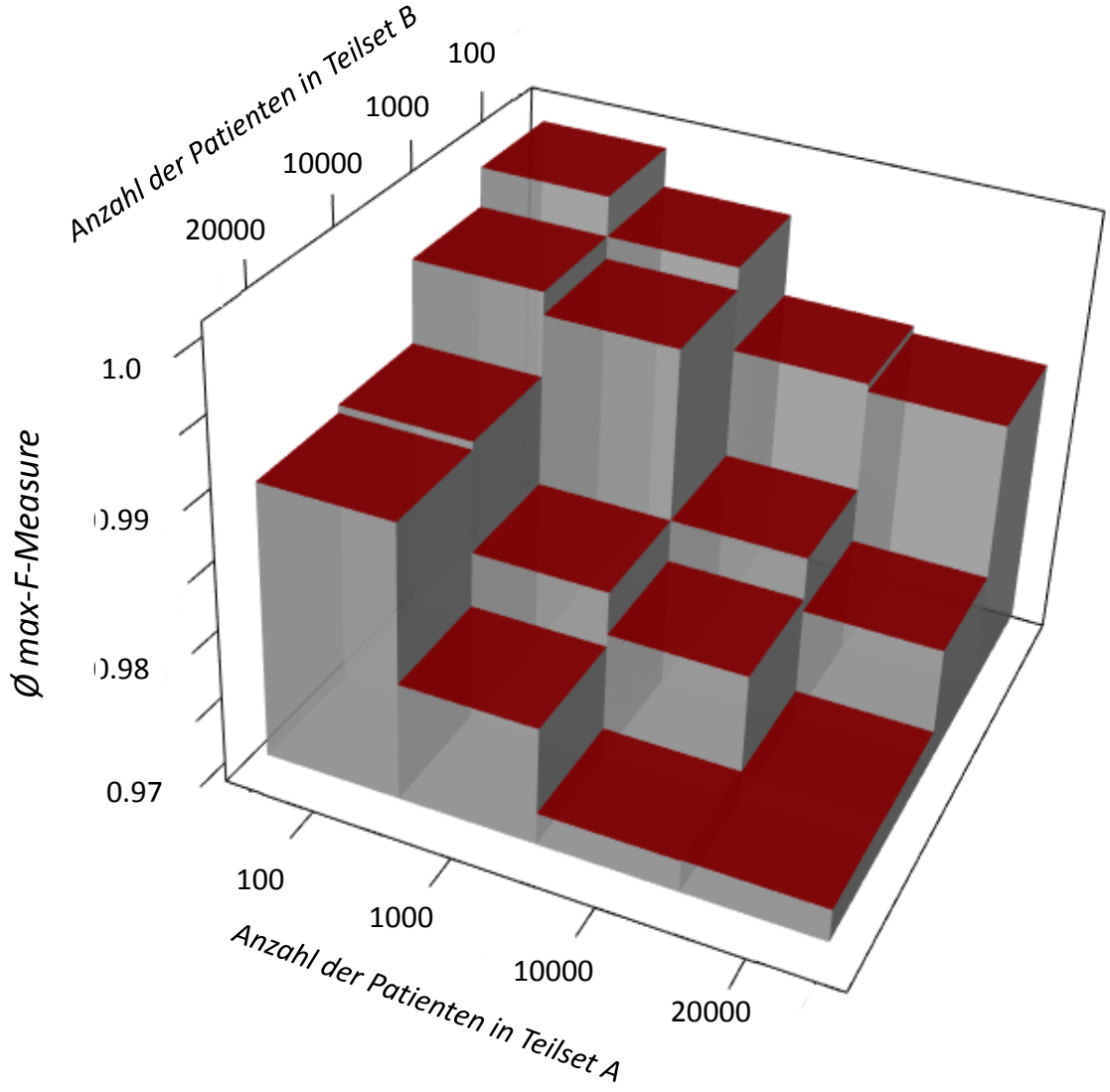


Abbildung 22: Gemittelter, maximaler F-Measure-Wert in Testsets mit spez. Größenkombination (3D).

Der Trend war relativ eindeutig: Bei schlechterer Datenqualität, also dem erhöhten Auftreten von Fehlern in Attributen zwischen echten Patientenübereinstimmungen sank der maximal erreichbare F-Measure-Wert. Eine schlechtere Datenqualität führte abhängig vom Fehler zu einer niedrigeren Gewichtung zwischen echten Übereinstimmungen. Damit konnte es passieren, dass echte Übereinstimmungen als falsche Links klassifiziert wurden, was in einer False-Negative-Bewertung resultiert hätte. Durch Transformationsfehler konnte es zudem zur Erhöhung des Gewichtes einer Nicht-Übereinstimmung kommen. Hierdurch entstanden

vermehrt False-Positives. Die Erhöhung beider Werte wirkte sich verringernd auf den F-Measure-Wert aus.

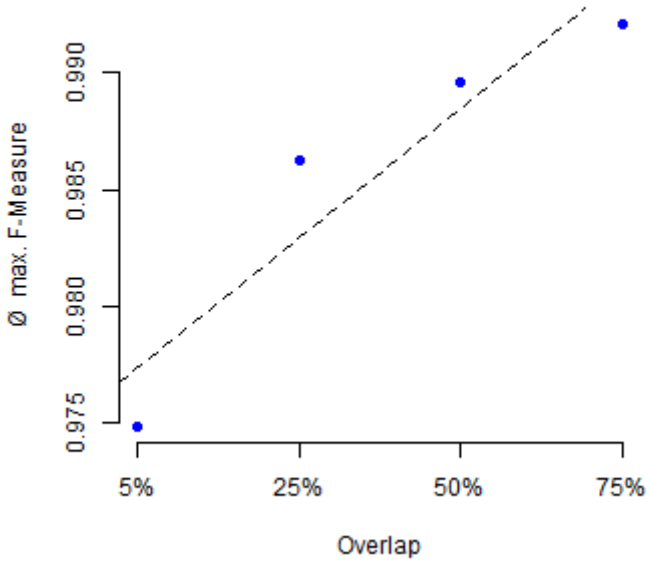


Abbildung 23: Gemittelter, maximaler F-Measure-Wert in Testsets abhängig von der Größe der Überlappung.

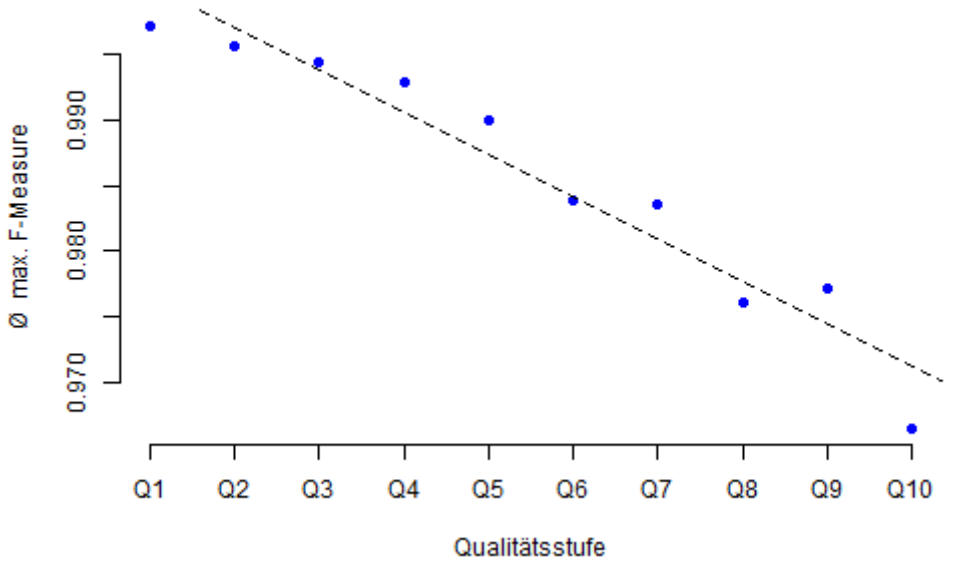


Abbildung 24: Gemittelter, maximaler F-Measure-Wert in Testsets abhängig von der Datenqualität.

3.2. Auf Trainingsset-Varianten basierende Klassifikationsergebnisse

Entsprechend *Kapitel 2.5* wurden 400 Template-Trainingssets erstellt, die zu jeweils einem der gegebenen Testsets in der Größe der Teilsets, der Größe des Überlappungsbereiches sowie der Fehlerhäufigkeiten übereinstimmten. Zudem wurde die Werteverteilung durch direktes Kopieren aus den Originaldaten weitestgehend identisch, mit Ausnahme des Überlappungsbereiches, übernommen. Zu den angesprochenen Template-Trainingssets wurden Trainingssetvarianten erstellt, die in jeweils einem der vier genannten Konstruktionsparameter von den Template-Trainingssets abwichen. Dies resultierte in 7 zusätzlichen Reihen von jeweils 400 Datensets. Zur Bekräftigung der Ergebnisse wurden jeweils 3 Serien dieser Sets sowie der Template-Trainingssets erstellt, was in insgesamt 9600 Datensets resultierte. Auf jedem dieser Trainingssets wurde ein Record-Linkage durchgeführt, auf der erhaltenen Gewichtsdatei wurde der jeweils optimale Klassifikator ermittelt (also derjenige, der den F-Wert maximiert) und die erhaltenen Klassifikatoren wurden letztendlich in die jeweils zugrunde liegenden Testsets eingepasst. Die Performanz der durch die Trainingssets erzeugten Klassifikatoren wurde anhand von F-Wert Berechnung an der gegebenen Position auf den jeweiligen Testsets bemessen und die ermittelten Werte wurden für weitere Auswertungen dokumentiert. *Abbildung 25* zeigt hierbei vergleichend die Performanz der verschiedenen Klassifikatoren nach Qualität der Testsets gruppiert. Die Kurve mit der Bezeichnung „Optimal“ beschreibt hierbei den maximal erreichbaren durchschnittlichen F-Measure auf den Testdaten, „Overlap (1-3)“ beschreibt hierbei die Klassifikationsgüte der Trainingssetvarianten mit einem festen Überlappungsbereich von (90%,30% sowie 3%), „Template“ beschreibt die Ergebnisse zur Klassifikationsgüte anhand der Template-Trainingssets, „Size (1-2)“ gibt die Klassifikationsgüte zu den Varianten mit konstanter Größe von 100 Patienten pro Testset bzw. halber Größe der original Testsets, Error bezeichnet die Ergebnisse die der Trainingssetvariante ohne Fehler zugrunde liegen und Distribution bezeichnet die Ergebnisse der Trainingssetvariante, bei der Wertausprägungen aus einer gleichverteilten Menge gezogen wurden.

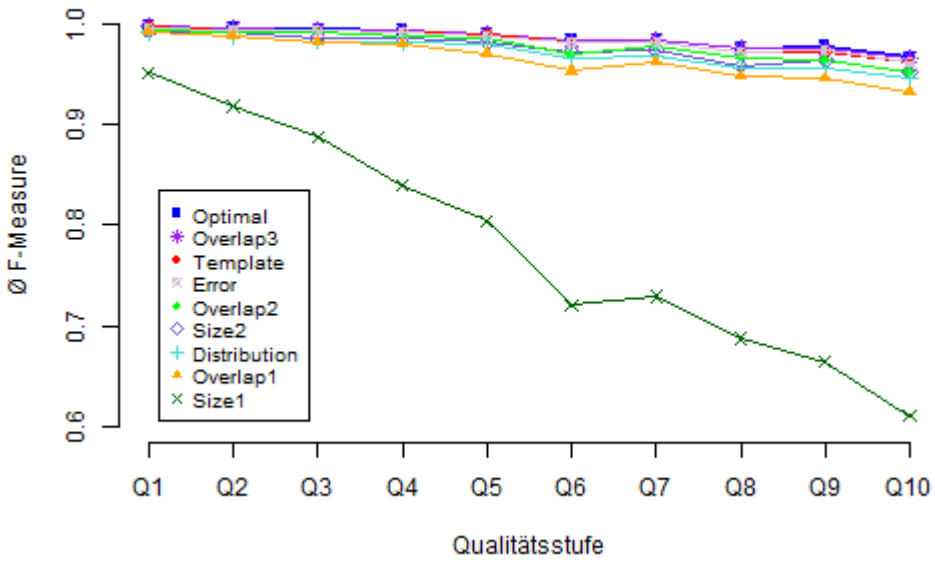


Abbildung 25: Gemittelte Klassifikationsgüte (F-Measure-Wert) von auf verschiedenen Trainingsset-Varianten basierenden Klassifikatoren, gruppiert nach Qualitätsstufe.

Die Grafik ist in der gegebenen Form nur schwer lesbar. Als eindeutiges Ergebnis zeigte sich jedoch schnell und eindeutig, dass die Trainingsset-Variante („Size1“), bei der die Größe der Teilsets auf 100 normiert wurde, nicht zur Klassifikation geeignet war. Die durchschnittlichen F-Measure-Werte lagen hierbei deutlich weit unter den Ergebnissen der anderen Klassifikatoren. Aus der nachfolgenden Grafik (*Abbildung 26*) wurde die letztgenannte Trainingsset-Variante entfernt und der Fokus richtete sich auf den Bereich der anderen Varianten

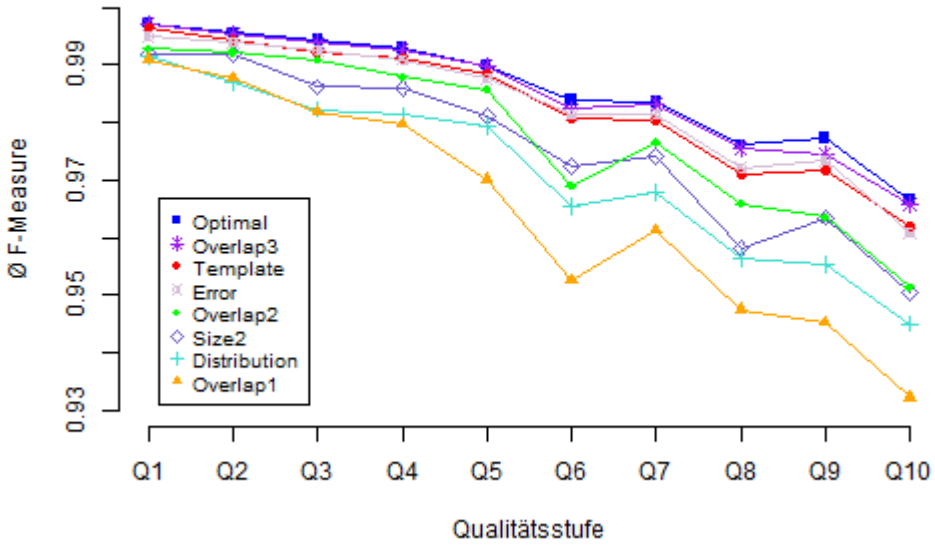


Abbildung 26: Gemittelte Klassifikationsgüte (F-Measure-Wert) von auf verschiedenen Trainingsset-Varianten basierenden Klassifikatoren, gruppiert nach Qualitätsstufe.

In gegebener Grafik zeigten sich nun deutlich die Unterschiede der einzelnen Trainingssetvarianten. Zwar war die Abweichung zwischen Template-Trainingsset und der zweiten Größenvariante („Size2“) nicht so extrem wie bei der ersten Variante, doch auch das Halbieren der Teilsetgrößen resultierte in vergleichsweise verminderten F-Werten. Beide Ergebnisse deuteten daraufhin, dass eine allgemeine Reduktion der Teilsetgrößen zu verminderten F-Werten führte. Dementsprechend sollte für einen optimalen Klassifikator, wie es bereits bei den Template-Trainingssets Usus war, die Teilsetgröße beibehalten werden.

Weiterhin wurde die Verteilung untersucht. Hierzu wurde nur eine Variante („Distribution“) geprüft, bei der die Ausprägungswerte in der Trainingssetvariante gleichmäßig verteilt wurden. Wie sich zeigte, führte die genannte Abweichung von der Originalverteilung ebenfalls zu einer relativ hohen Verminderung der Klassifikationsgüte.

Grundsätzlich überschnitten sich die Konzepte der Größenvariation und der Verteilungsvariation, da sich eine Anpassung der Größe meist direkt auf die Verteilung auswirkte. Dass eine Abweichung der Verteilung beim probabilistischen Record-Linkage direkten Einfluss auf die Klassifikation hatte, war aber grundsätzlich nachvollziehbar. Schließlich basierte beim probabilistischen Record-Linkage die Gewichtsberechnung auf den unter *Kapitel 1.3.2* beschriebenen u-Werten, die sich direkt aus der Häufigkeit von Ausprägungswerten ableiteten.

Die Ergebnisse zum Overlap-Parameter lieferten neue Erkenntnisse. Hierbei wurden drei Varianten geprüft (90% (“Overlap 1“), 30% (“Overlap2“), 3% (“Overlap3“)). Wie sich zeigte, verbesserte sich die Klassifikationsgüte bei jeder Verminderung der Größe des Überlappungsbereiches. Da die Versuche jeweils, wie bereits erwähnt, dreimal wiederholt wurden und sich jeweils dasselbe Bild zeigte, waren Zufallsergebnisse zu hoher Wahrscheinlichkeit auszuschließen. In der Variante mit 3% Überlappungsbereich konnte sogar die Güte des Template-Klassifikators übertroffen werden. Die ursprüngliche Hypothese, dass eine maximale Anpassung des Überlappungsbereiches an die originalen Testdaten zu einer optimalen Klassifikation führt, wurde somit widerlegt. Vielmehr zeigte sich, dass ein möglichst kleiner Überlappungsbereich der Klassifikation dienlich war. Wie schon die Größe wirkte sich auch die Veränderung der Überlappung auf die Werteverteilung aus. Je größer der Überlappungsbereich gewählt wurde umso mehr Orginal-Patienten wurden aus Teilset B entfernt und umso mehr Kopien wanderten von Teilset A nach Teilset B. Die kopierte Menge aus Teilset A und deren Werteverteilung lag also überrepräsentiert vor, wohingegen Werte aus Teilset B verlorengingen. Die Veränderung der Verteilung beeinflusste, wie bereits beschrieben, die u-Werte und konsequenterweise die finale Gewichtsberechnung und Klassifikation.

Eine weitere neue Erkenntnis war das Ergebnis, dass Fehlerraten zur Vorhersage eines optimalen Klassifikators nicht unbedingt benötigt waren. Wie die Variante „Error“ in *Abbildung 26* zeigte, gab es quasi keinen Unterschied zwischen der Klassifikationsqualität zu auf den Template-Trainingsset basierenden Klassifikatoren, bei denen Fehlerhäufigkeiten im Überlappungsbereich mit denen aus den Testdaten übereinstimmten. Eine ursprüngliche Vermutung war es, dass eine Berücksichtigung der Fehler gerade bei Testsets niedrigerer Datenqualität zu einer Verbesserung der Qualität führen würde, doch dies konnte anhand von Abbildung 26 widerlegt werden. Die Interpretation der Hypothese, an der sich die Konstruktion der Template-Trainingssets orientierte, konnte also ein zweites Mal widerlegt werden.

Analog zu *Kapitel 3.1* wurden aus Gründen der Vollständigkeit noch die Abbildung (*Abbildung 27 sowie Abbildung 28*) der durchschnittlichen F-Werte bei Gruppierung nach Teilsetgrößen bzw. Überlappung nachgereicht. Deren Ergebnisse deckten sich mit den unter *Kapitel 3.1* vorgestellten Beobachtungen.

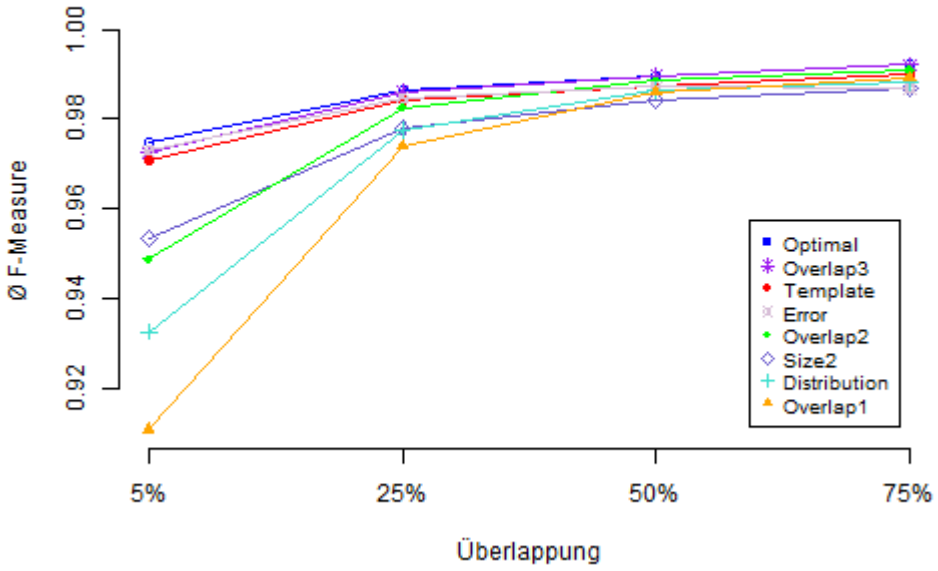


Abbildung 27 : Gemittelte Klassifikationsgüte (F-Measure-Wert) von auf verschiedenen Trainingsset-Varianten basierenden Klassifikatoren gruppiert nach Größe des Überlappungsbereiches.

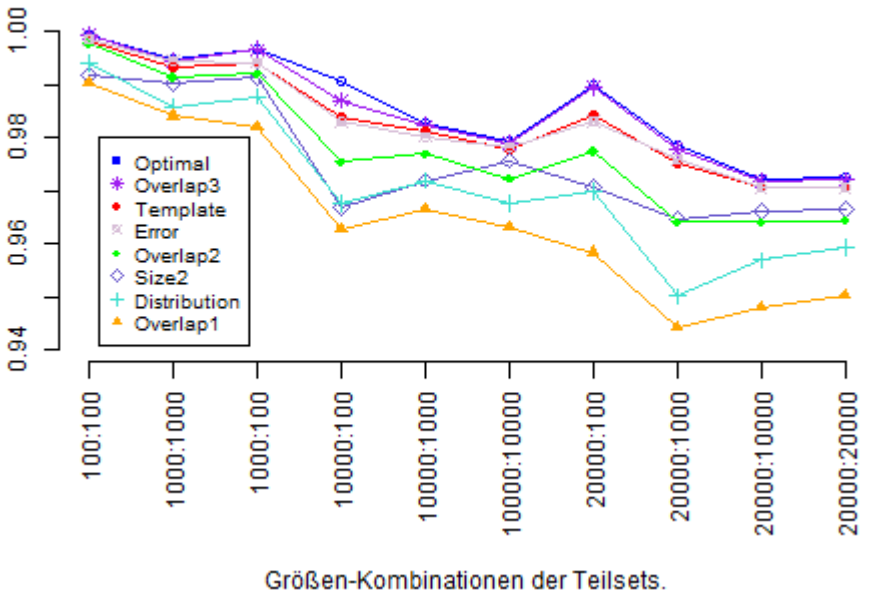


Abbildung 28 Gemittelte Klassifikationsgüte (F-Measure-Wert) von auf verschiedenen Trainingsset-Varianten basierenden Klassifikatoren gruppiert nach Größe der Teilsets.

3.3. **CLARA**

Basierend auf den vorgehenden Auswertungen war es möglich, die ursprüngliche Interpretation der Hypothese aus *Kapitel 2.4* zu widerlegen und es konnten neue, optimierte Empfehlungen zu den Konstruktionsparametern formuliert werden, die die Klassifikationsqualität im Vergleich zur Template-Variante übertrafen. Hierauf basierte das neu entwickelte CLARA-System. CLARA stand hierbei für ***CLA****ssification for* ***R****ecord-Linkage with* ***A****rtificial Trainingssets*. *Tabelle 12* beschreibt die optimierte Konstruktionsparametrisierung des CLARA-Systems im Vergleich zur Konstruktion der zuvor beschriebenen Template-Trainingssets.

Tabelle 12: Beschreibung der Parametrisierung der Konstruktion von Trainingssets des CLARA Systems.

Konstruktions-Parameter	**Konfiguration (Template)**	**Konfiguration (CLARA)**
Größe der Teilsets	Identische Größenverhältnisse der Teilsets zum zugrunde liegenden Testset.	Entsprechend Template-Trainingsset-Konstruktion
Größe des Überlappungsbereiches	Identisch zur Größe des Überlappungsbereiches des zugrunde liegenden Testsets.	Möglichst minimal, jedoch ausreichend groß um eine Klassifikation grundsätzlich zu erlauben. Für diese Arbeit und generell als Richtwert werden 3% der Größe des jeweils kleineren Teilsets vorgeschlagen.
Verteilung	Trainingsset A identisch zu Testset A. Trainingsset B bis auf Überlappungs-bereich identisch zu Testset B.	Entsprechend Template-Trainingsset-Konstruktion
Fehlervorkommen	Häufigkeitswerte zu Fehlervorkommen stimmen mit denen des Testsets überein.	Es werden keine Fehler in den Überlappungsbereich eingebracht.

Das CLARA System war hierbei von den genauen Angaben der Parametrisierung, die zuvor über den Dateinamen übergeben wurden, unabhängig und konnte hierdurch automatisiert im Praxiseinsatz verwendet werden. Die Größe der Teilsets ließ sich auch ohne Vorkenntnisse aus den originalen Testdaten auslesen. Schätzungen der Größe des Überlappungsbereiches waren

nicht mehr notwendig, da ein konstanter Wert (3%) verwendet wurde. Ebenso waren Schätzungen zu den Fehlerraten unnötig, da diese nach den Ergebnissen aus *Kapitel 3.2* nicht mehr benötigt wurden, bzw. der Klassifikation nicht zugute kamen. Man versuchte die Verteilung, wie gehabt, möglichst unverändert zu belassen, was ohne Vorkenntnisse, wie bereits beschrieben, durch einfaches Kopieren aus den Originaldaten möglich war.

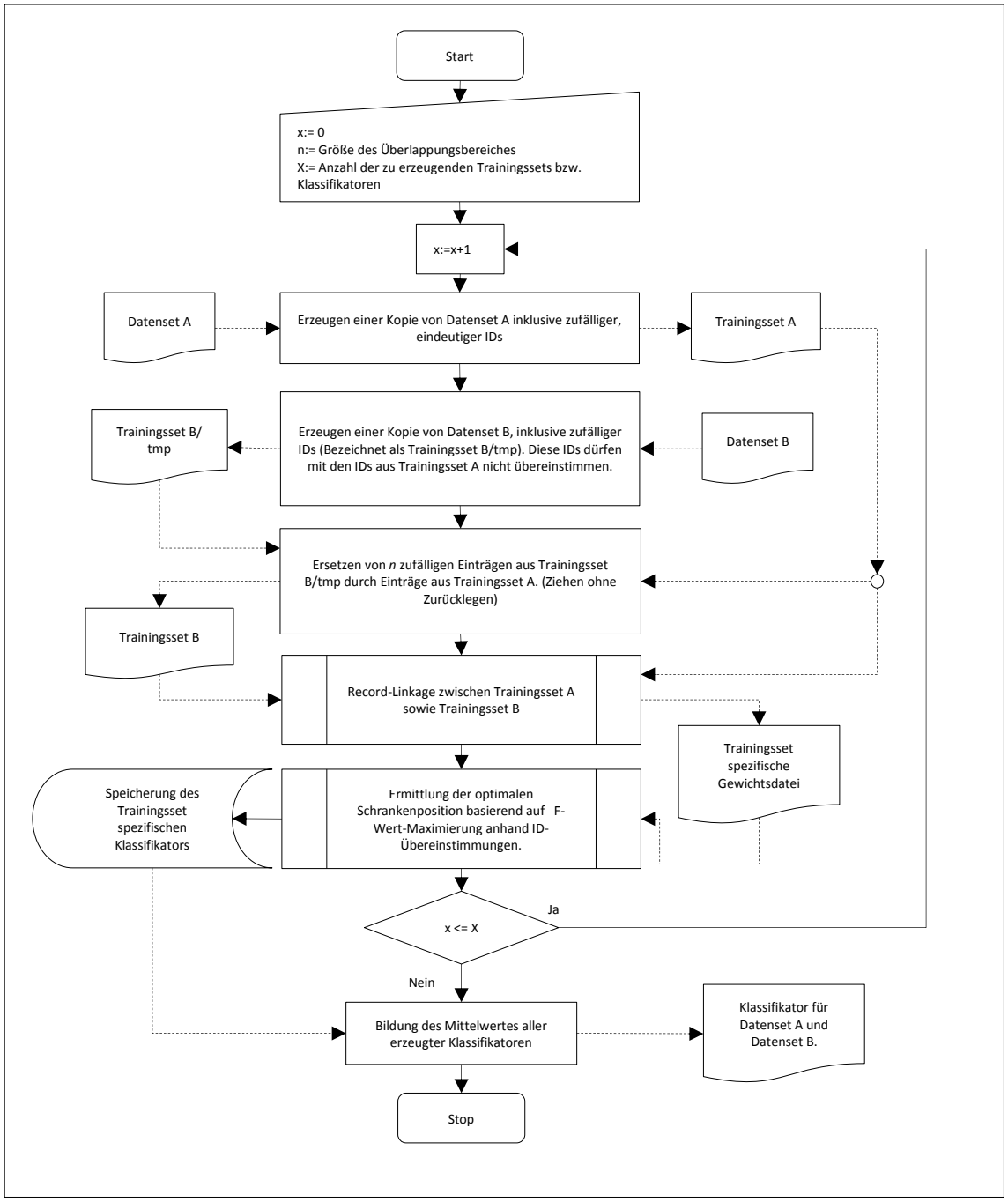


Abbildung 29: Schematischer Ablauf der ganzheitlichen CLARA-Methodik.

Durch Erzeugung und Schrankenberechnung mehrerer Trainingssets mit Variation im Überlappungsbereich konnten mehrere CLARA-Klassifikationen zu jeweils einem Testset hervorgesagt werden. Ein Mittelwert dieser multiplen Klassifikatoren würde also den maximal möglichen Fehler, also die Abweichung des Klassifikators vom eigentlichen optimalen Schrankenwert, minimieren, was beim konkreten Einsatz beachtet werden sollte. *Abbildung 29* beschreibt zusammenfassend den schematischen Ablauf des finalen CLARA-Verfahrens.

3.4. Vergleich verschiedener Klassifikationsmethoden

Basierend auf den Ergebnissen aus *Kapitel 3.2* wurde das CLARA-System, das im Methodenteil dieser Arbeit als parameter-optimierte Variante bezeichnet wurde, modelliert. Beim CLARA-System handelte es sich um ein System zur Konstruktion von Trainingsdaten anhand gegebener Originaldaten, die anschließend zu überwachter Klassifikation verwendet werden konnten. Ob sich das System auch für den Realeinsatz geeignet ist und ob es mit anderen, ausgewählten Klassifikationsmethoden konkurrieren kann, wurde über eine Reihe von Methodenvergleichen geprüft (siehe *Kapitel 2.6*). Bei den verglichenen Methoden handelte es sich um CLARA, Single-Linkage-Clustering, zwei Varianten des SNN-Algorithmus mit Variation in der Keimmenge sowie manuelle Klassifikation anhand von Histogrammen entsprechend dem Vorgehen in der DKFS.

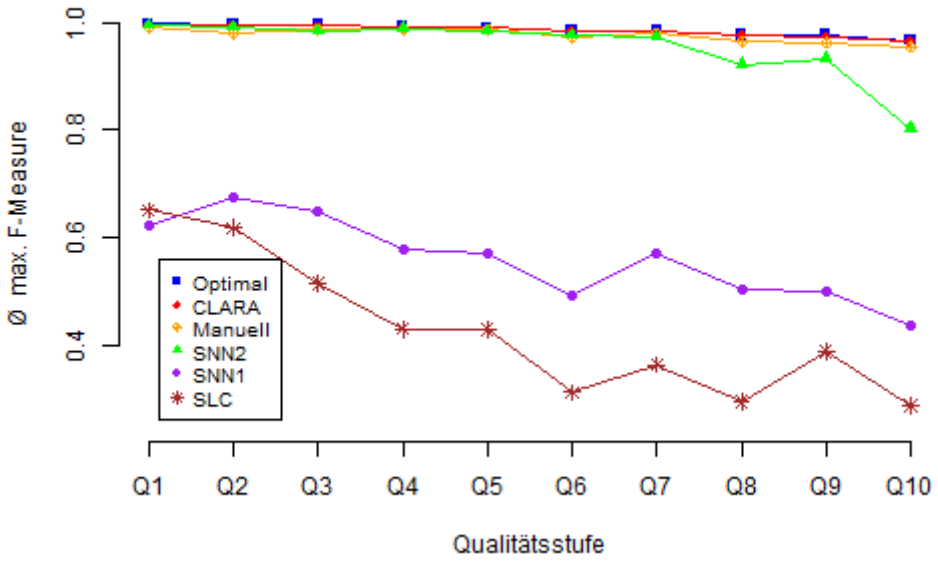


Abbildung 30: Durchschnittlicher F-Measure-Wert verschiedener Klassifikatoren abhängig von der Datenqualitätsstufe.

Abbildung 30 beschreibt die Abgleichsgüte der verschiedenen geprüften Klassifikationssysteme abhängig von der Qualität der zugrunde liegenden Testsets. In dieser sowie den nachfolgenden Grafiken bezeichnen die Kürzel „Optimal“ den maximal erreichbaren durchschnittlichen F-Measure auf dem zugrundeliegenden Testset, „Clara“ steht für die Klassifikationsgüte von CLARA, „Manuell“ beschreibt die Klassifikationsgüte basierend auf manueller Schrankenfindung wohingegen „SNN(1-2)“ die Ergebnisse des SNN mit Keimmengenbestimmung entsprechend Formel 17 sowie Keimmengenbestimmung anhand festen Treshholds beschreibt. „SLC“ steht weiterführend für die Ergebnisse des Single-Linkage-Clusterings. Es zeigten sich hierbei zwei Gruppen von Klassifikatoren. Die Klassifikatoren mit einem F-Measure-Wert oberhalb von 0,95 erschienen als für den Realeinsatz verwendbar, wohingegen die beiden verbleibenden Klassifikatoren weit unterhalb dieses Wertes lagen und für die Klassifikation im Record-Linkage als eher ungeeignet zu bewerten waren. Beim SLC, das nicht unbedingt auf das Konzept des Record-Linkage optimiert wurde, war dies noch nachvollziehbar, bei der ersten SNN-Variante überraschte dies allerdings. Es zeigte sich, dass hierbei die Auswahl der korrekten Keimmenge eine immense Rolle auf die finale Abgleichsgüte spielte. Die Keimmenge der ersten Variante des SNN wurde anhand einer empfohlenen Formel aus der Originalpublikation erzeugt, die das Konzept des SNN vorstellt [71]. Es schien, als würden die durch diese Formel erzeugten Keimmengen zu klein erstellt, weswegen die gegebene Klassifikation oft in den Randbereichen der Gewichtsdateien fehlerhafte Schranken vorschlug und sich demnach kaum von der Klassifikationsgüte des SLCs unterschied. Im SNN2 wurden die Keimmengen manuell anhand von Treshholds, also festen Schrankenwerten erstellt. Die Bereiche wurden größer gewählt, wodurch die Klassifikationsschranken nicht fälschlicherweise in die Randbereiche eingepasst wurden, da diese bereits in den Keimmengen enthalten waren. Hierdurch konnte eine immense Steigerung der Abgleichsqualität erzielt werden. Als Fazit ließ sich sagen, dass die SNN Methode nur in einer Variante brauchbare Ergebnisse erzielen konnte. Die Auswahl der Keimmenge war demnach ein Unsicherheitsfaktor, der die komplette Klassifikation kompromittieren konnte. Nicht nur aufgrund dieses Unsicherheitsfaktors, sondern auch aufgrund der komplexen und anspruchsvollen Implementierung wäre Benutzern, die sich nicht tiefer mit der Methodik befassen, sondern diese lediglich nutzen wollen, abzuraten. *Abbildung 31* beschränkt sich nun auf die Klassifikatoren abzüglich der ersten Variante des SNNs sowie des SLCs.

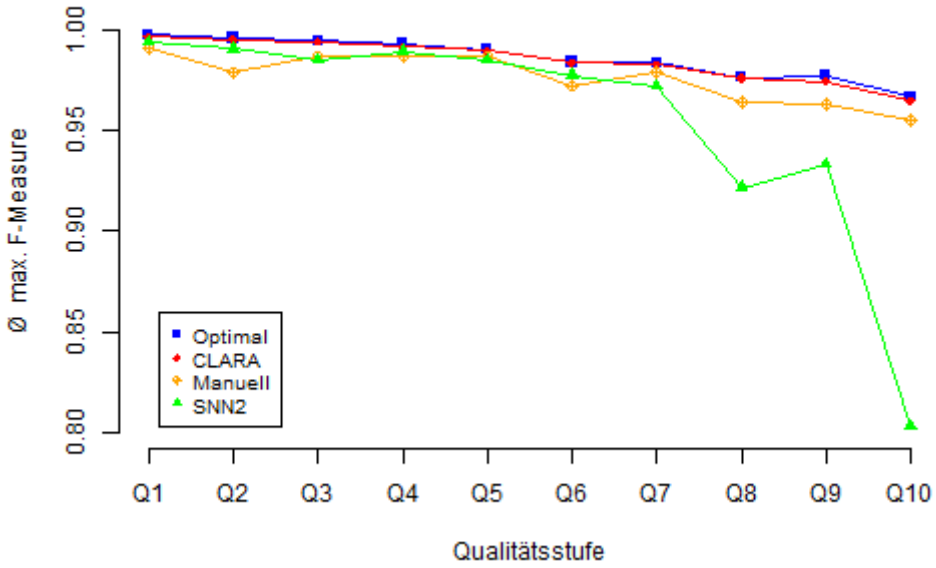
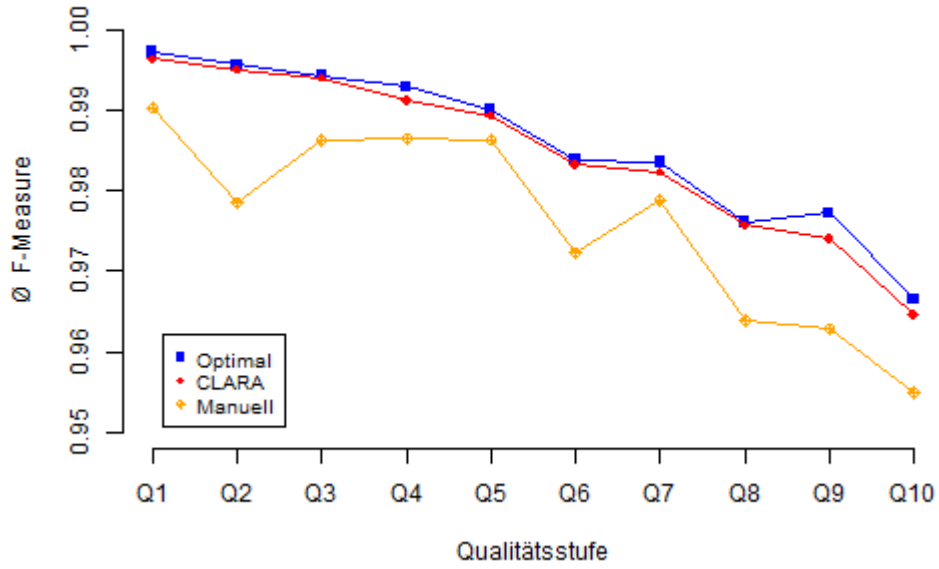


Abbildung 31: Durchschnittlicher F-Measure-Wert verschiedener Klassifikatoren abhängig von der Datenqualitätsstufe.

Hierbei unterschied sich vor allem der SNN in der zweiten Variante von den restlichen Methoden. Auf Testsets mit einer Qualitätsstufe einschließlich dem Wert Q6 erzeugte der Klassifikator noch gute Ergebnisse, erzielte dabei sogar teils bessere Ergebnisse als die manuelle Klassifikation, brach jedoch ab einem Wert von Q7 in Bezug auf die Abgleichsqualität



stark ein.

Abbildung 32 Durchschnittlicher F-Measure-Wert verschiedener Klassifikatoren abhängig von der Datenqualitätsstufe.

Im direkten Vergleich übertraf die manuelle Klassifikation den SNN. Im Vergleich zum CLARA-System zeigte sich vor allem, dass der SNN nicht nur bei schlechter Datenqualität schlechter als das CLARA-System abschnitt, sondern auch bei guter Datenqualität.

Hiermit verblieb noch ein direkter Vergleich zwischen CLARA und der manuellen Klassifikation, der in *Abbildung 32* dargestellt wird. Wie sich zeigte, lag CLARA jederzeit über den manuellen durchschnittlichen Schätzwerten der Schrankenbestimmung. Generell lag die Klassifikation meist sogar nur sehr knapp unter den maximal erreichbaren F-Werten, die bei einer perfekten Klassifikation möglich gewesen wären. Die Datenqualität wirkte sich hierbei nicht wie beim KNN negativ auf das Klassifikationsergebnis aus.

4. Diskussion

4.1. Begründung der Konzeption eines überwachten Klassifikationssystems

Im Rahmen der DKFS wurden wissenschaftlich offene Fragestellungen und Probleme in Bezug auf die Klassifikation im Bereich des Privacy-Preserving-Record-Linkage identifiziert. Aufgrund schwieriger Datenverhältnisse, wie sie sich gerade im Fall der Daten von Angehörigen präsentierten, kann es Probleme bereiten, eine passende Klassengrenze bzw. einen binären Klassifikator zu bestimmen [67].

Zu Problemen dieser Art gibt es nur wenig Literatur, da zum einen wohl die Datengrundlage in vielen Projekten eine einfachere Klassifikation erlaubt. Zum anderen scheint es, als würde die Relevanz der Klassifikation oft im Schatten der Gewichtsberechnung stehen, die in wissenschaftlicher Literatur die meiste Aufmerksamkeit genießt.

Bei automatisierten Klassifikationsmethoden, die während eines anonymen Record-Linkage alternativ zum manuellen Vorgehen anwendbar wären [71], handelt es sich primär um regelbasierte, überwachte sowie unüberwachte Klassifikationssysteme. Während regelbasierte Klassifikationsmethoden meist sehr projektspezifisch aufgesetzt werden, konzentrierten sich die Untersuchungen der Klassifikationsmethoden in dieser Arbeit dagegen vorrangig auf den Vergleich zwischen unüberwachter sowie überwachter Klassifizierung [38,41,71,80].

Insbesondere wurde dabei eine eigens entwickelte, schon früh entworfene Idee zur überwachten Klassifizierung ausgearbeitet, die später mit anderen Klassifikationsmethoden verglichen wurde. Die Fokussierung auf die überwachte Klassifizierung rührte aus der Annahme, dass schlechte Datenqualität eine überwachte Klassifikation weniger negativ beeinflussen sollte als eine unüberwachte Klassifikation, die bei Artefakten in der Gewichtsmenge, wie etwa unerwartete, zufällig auftretende Gewichtssprünge, immer die Gefahr einer kompletten Fehlklassifikation birgt. Aufgrund der Tatsache, dass die manuelle Klassifikation auf Histogramm-Daten ebenfalls dieselben Probleme aufweist – also Anfälligkeit gegenüber Datenartefakten – stellte sich die überwachte Klassifikation als unabhängige Variante hierzu dar [42].

Zwar existieren auch im Bereich des Record-Linkage Ansätze zu überwachter Klassifikation [38,83,84], allerdings fehlen hier eindeutige Anweisungen bzgl. Parametrisierung und Auswahl

der zugrunde liegenden Trainingssets. Bezüglich des neuen Ansatzes gab es deswegen das Ziel, eine möglichst einfache und eindeutige Anwendung zu erlauben, die im Grunde genommen keine externen Trainingsdaten voraussetzte, sondern die Trainingsdaten direkt aus der zugrunde liegenden Testdatenmenge generierte. Dreh-und Angelpunkt dieser Arbeit war daher, ein derartiges System aufzusetzen und auf verschiedenen Testsets auf die Abgleichsgüte zu prüfen.

4.2. **Zugrundeliegende Arbeitsmaterialien**

Analysen im Bereich des Record-Linkage sind schwierig, da es an guten externen Testdaten mangelt [82]. Aus diesem Grund wurde anhand von Klinikumsdaten eine umfangreiche Menge von insgesamt 400 Testdatensätzen konzipiert, die sich in verschiedenen Parametern, der Größe, dem Überlappungsbereich als auch der Datenqualität unterschieden. Somit war eine Prüfung von Methoden, die im Bereich des Record-Linkage angesiedelt sind, unter vielen verschiedenen Testbedingungen möglich. Während z.B. Testdaten der Qualitätsstufe 1-2 eine sehr gute Datenqualität widerspiegelten, entsprachen Testdatensätze der Qualitätsstufe 8-10 eher schwierigen Datenverhältnissen mit vielen fehlenden Werten und auftretenden Fehlern in den einzelnen Ausprägungen der Patienteneinträge.

Zu jedem Testdatensatz wurde ein probabilistisches Record-Linkage durchgeführt, wodurch jeweils eine Gewichtsdatei für vergleichende Analysen erzeugt wurde. Das verwendete System entsprach hierbei in Bezug auf die Abgleichsgüte (Sensitivität/Spezifität) anderen aus verschiedener Literatur bekannten Angaben (siehe *Tabelle 13 sowie Abbildung 33/Abbildung 34*).

Tabelle 13: Angaben zu Spezifität und Sensitivität bzgl. probabilistischem Record-Linkage.

Quelle	Kurzbeschreibung	Spezifität	Sensitivität
Boonchai et al. [101]	Für eine Prüfung der Qualität eines Record-Verfahrens zwischen zwei künstlichen Datenbanken wurden einwegverschlüsselte Kontrollnummern anhand von Personen-identifizierenden Daten aus verschiedenen Quellen erzeugt und zu Datenbank-Einträgen zusammengefügt.	100%	95%-100%
Durham et al. [53]	Record-Linkage auf 756.629 künstlichen Patienten-Daten, ausgehend von 100.000 realen Patienten mit einem Überlappungsbereich von 0.01 %.	~100%	~97%

Quelle	Kurzbeschreibung	Spezifität	Sensitivität
Contiero et al. [102]	Es wurde ein Abgleich auf einem Teil von Patientendaten des französischen Krebsregisters der Lombardie (20.724 Einträge) mit Daten zu sozialer Sicherheit durchgeführt (1.021.846 Einträge) durchgeführt. Die Ergebnisse wurden über manuelle Kontrolle, also nach Golds-Standard ausgewertet.	98.8%	96.5%
Fonseca et al. [103]	Die nationale, brasilianische HIV/AIDS Überwachungsdatenbank (559.442 Einträge) wurde gegen eine Menge von 6.444.822 Daten zu registrierten Toden abgeglichen.	99.6%	87.6%
Migowski et al. [104]	In dieser brasilianischen Studie wurde versucht, die Qualität des Record-Linkage abzuschätzen, indem in einer Datenbank zu verstorbener Bevölkerung nach am Herzen operierten Patienten gesucht wurde.	100%	90.6%
Quantin et al. [19]	Abgleich von manueller und automatischer Methodik im Burgundy-Register von Patientendaten mit zum Verdauungssystem assoziierten Krebsarten.	97%	93%
Fournel et al. [105]	Abgleich des größten französischen Krebsregisters und Todesfällen in Frankreich zwischen 1998–2004.	99.5%	94.8%
Silveira et al. [75]	Review verschiedener Paper und Studien in Bezug auf Abgleichsqualität von probabilistischem Record-Linkage.	99-100%	74-98%

Wie *Abbildung 32* und *Abbildung 33* demonstrieren, übertrafen die Werte zu Sensitivität und Spezifität abhängig von der Qualitätsstufe meist sogar die gegebenen Vergleichswerte. Bei Nennung mehrerer Werte in der jeweiligen Arbeit wurde innerhalb der angegebenen Grafiken ein Mittelwert angegeben. Berücksichtigt werden muss hierbei allerdings, dass für das eigene System eine optimale binäre Klassifikation, sowie das Bekanntsein der zugrunde liegenden Häufigkeiten der m-Werte verwendet wurden, was im Realeinsatz nicht der Fall ist und wodurch, mit hoher Wahrscheinlichkeit, eine verbesserte Abgleichsqualität erreicht werden konnte.

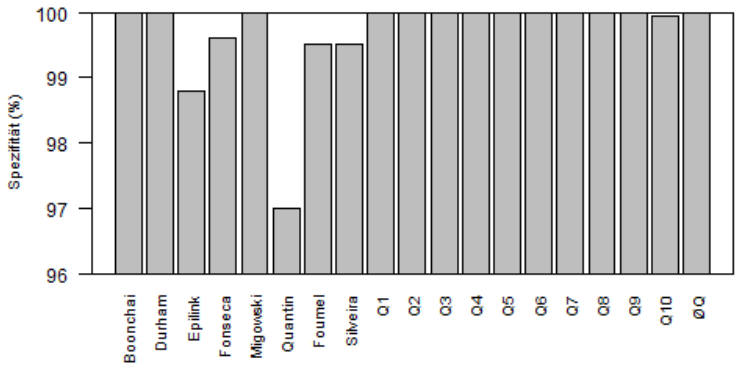


Abbildung 33: Vergleich der veröffentlichten Spezifitätswerte von probabilistischen Record-Linkage-Methoden aus verschiedenen Literaturquellen mit Mittelwerten des Matchings in dieser Arbeit auf Testsets gruppiert nach Qualitätsstufe.

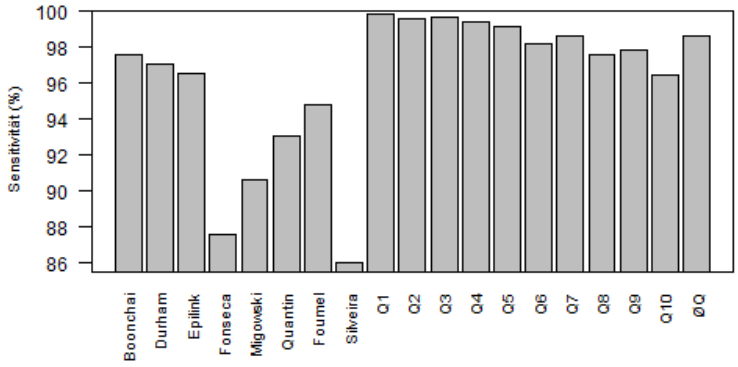


Abbildung 34: Vergleich der veröffentlichten Sensitivitätstswerte von probabilistischen Record-Linkage-Methoden aus verschiedenen Literaturquellen mit Mittelwerten des Matchings in dieser Arbeit auf Testsets gruppiert nach Qualitätsstufe.

4.3. **Hypothese als Ausgangspunkt des wissenschaftlichen Vorgehens**

Bei der Konzipierung des neuen überwachten Klassifikationsansatzes wurde schließlich initial eine Hypothese aufgestellt, die besagte: Je ähnlicher zwei Datensets sind, umso ähnlicher sind auch ihre Klassifikatoren. In diesem Kontext musste Ähnlichkeit definiert werden und anhand dessen wurde ein Template-Trainingsset konzipiert, das mit dem jeweiligen Testset in Größe der Teilsets, Größe des Überlappungsbereiches, sowie Häufigkeit der Fehlerraten übereinstimmte. Zudem wurde versucht, auch die Werteverteilung möglichst gut zu übernehmen, um die Ähnlichkeit zu maximieren. Es ist nicht auszuschließen, dass es hierbei

Ansätze gibt, die zu einer noch höheren Ähnlichkeit zwischen Test- und Trainingsdaten führen würden.

Der Hypothese folgend müsste also ein optimaler Klassifikator auf diesem Template-Trainingsset, kalibriert am maximalen F-Measure-Wert, auch auf dem zugrunde liegenden Testdatenset eine Klassifikation mit hoher F-Measure-Bewertung erzeugen. Widersprüchlich wäre es also gewesen, wenn Trainingsdaten, die nicht diesen Ähnlichkeitsanforderungen entsprächen, zu besseren Klassifikationsergebnissen geführt hätten. Um die Annahme also zu prüfen, wurden zu den 400 Testdatensets insgesamt jeweils 7 weitere Trainingsdaten-Varianten aufgesetzt, die sich jeweils in einem Parameter, entweder der Größe der Teilsets, der Größe des Überlappungsbereiches, den Fehlerhäufigkeiten, oder der Werteverteilung von den gegebenen Template-Trainingsset unterschieden.

Die ursprüngliche Hypothese wurde dabei widerlegt. Es zeigte sich, dass es zwar galt, Größe und Verteilung so gut wie möglich beizubehalten, dass jedoch Übereinstimmung des Überlappungsbereiches zu keiner Verbesserung der Klassifikation führte, sondern im Gegenteil sogar zu einer Verschlechterung. Gemäß den Analysen sollte der Überlappungsbereich, der bei der Methodik mit neuen Werten belegt wird, möglichst klein gewählt werden. In dieser Arbeit wurden 3% der Größe des kleineren Trainingssets empfohlen, um die Werteverteilung möglichst minimal zu beeinflussen. Sicherlich waren auch andere Werte hierzu denkbar. Es musste lediglich vermieden werden, dass der Überlappungsbereich komplett oder nahezu leer verblieb. Die generelle Aussage lautet, je kleiner der Überlappungsbereich umso besser das Klassifikationsergebnis, jedoch darf der Überlappungsbereich hierbei nicht leer sein. Auch ein Überlappungsbereich von lediglich einem oder ein paar Links hätte zu Problemen führen können. Der exakte Empfehlungs-Wert ist hierbei grundsätzlich nicht fest spezifizierbar, sollte also als Kritikpunkt und Unsicherheit der Technik im Hinterkopf behalten werden.

Wie sich zudem zeigte, spielten auch die Häufigkeiten der Fehler in den Überlappungsbereichen keine entscheidende Rolle. Diese beeinflussten die Klassifikation weder positiv noch negativ.

Auf die Konstruktion eines optimierten Trainingsdatensets wirkt sich dies natürlich positiv aus, da weder Überlappungsbereich, noch Fehler korrekt abgeschätzt werden müssen. Hätte sich herausgestellt, dass diese Parameter denen der Ursprungsdaten entsprechen müssten, wäre die Umsetzung einer Anwendung im Realeinsatz deutlich schwieriger gewesen, da man dann Schätzwerte zu diesen Parametern benötigt hätte. Im Grunde genommen wäre dies das Henne-Ei-Problem, bei dem Werte, die man eigentlich bestimmen will (z.B. die Größe des

Überlappungsbereichs) im Vorfeld bestimmt werden müssten. Die optimierte Variante bedient sich nun allerdings lediglich der gegebenen Größen der Teilsets der Originaldaten, der Verteilungswerte zuzüglich eines zufälligen Überlappungsbereiches, sowie einem konstanten, niedrigen Wert für die Größe des Überlappungsbereichs. Diese vollautomatisierbare Technologie wurde CLARA benannt.

4.4. Abgleich und Bewertung verschiedener Klassifikatoren

Um den ursprünglichen Gedanken zu bestätigen, dass überwachte Klassifizierer gerade auf Testdaten mit schlechter Datenqualität im Vergleich zu den unüberwachten Systemen überlegen klassifizieren und um die Klassifikationsgüte von CLARA zu bewerten, wurde das System mit Algorithmen der unüberwachten Klassifizierung verglichen. Neben einer einfachen, aus dem maschinellen Lernen bekannten Clustering-Methode wurde hierbei das System auch mit einem auf das Record-Linkage ausgelegten Klassifikator, einem zweistufigen KNN mit vorhergehender Bestimmung einer Keimmenge aus dem Bereich des Aktiven-Lernens, dem SNN, verglichen. Basierend auf Tests übertrifft der zuletzt genannte Algorithmus andere unüberwachte Klassifikationssysteme [71] wie beispielsweise den hochgelobten TAILOR-Klassifikator [93].

Wie sich zeigte, schnitt der Clustering-Algorithmus, also das SLC, erwartungsgemäß schlecht ab. Naive Clustering-Algorithmen suchen prinzipiell nach besonderen Punkten, wie beispielsweise größeren Abständen in der Datengrundlage, und verwenden diese als Schrankenanker für die Klassifikation. Da diese Punkte oftmals gerade an den Rändern einer Gewichtsmenge vorkommen, sind die einfachen Clustering-Methoden also eher ungeeignet. Der SNN-Algorithmus konnte hingegen auf Daten mit hoher Datenqualität sehr gute Klassifikationsergebnisse, die nahe an der maximal möglichen Klassifikationsqualität lagen, erzielen. Einschränkend wäre hierbei zu nennen, dass die Klassifikationsgüte von der korrekten Auswahl der Keimmenge abhängt. Hierzu wurden zwei Varianten geprüft, wobei die eine der anderen stark überlegen war. Eine derartige Unsicherheit bei der Konfiguration eines Systems ist anwenderunfreundlich und benötigt ein gewisses Maß projektspezifischen, bzw. wissenschaftlichen Know-Hows. Solche Unsicherheiten sind bei CLARA nicht gegeben – die Anwendung ist bis auf die Festlegung der Größe des Überlappungsbereiches, für die ein konstanter Empfehlungswert erstellt wurde, eindeutig.

Wie bereits ausgeführt, konnte die zweistufige Methodik gute Ergebnisse auf Testdaten mit hoher Datenqualität erzielen. Auf Testsets mit mangelnder Datenqualität nahm die Güte der

Klassifikation jedoch rapide ab, da sich mit Abnahme der Datenqualität auch die Häufigkeit von Datenartefakten (unerwartete Abstände, Anhäufungen) erhöht. CLARA übertraf die Klassifikation des genannten Klassifikators auf niedriger Datenqualität bei Weitem, überraschender Weise zeigte sich aber, dass CLARA auch auf Datensätzen mit hoher Datenqualität ähnliche bzw. sogar bessere Ergebnisse als der SNN erzielte.

CLARA offenbart sich hierbei also als das System mit der besseren und von der Datenqualität unabhängigen Klassifikationsgüte. Zumal die Konfiguration einfach und eindeutig ist, stellt sich CLARA bezüglich der untersuchten Testdaten als das überlegene System dar. Die Laufzeit wurde während des Projektes nicht dokumentiert, doch auch hier scheint CLARA keine größeren Probleme zu bereiten. Die Konstruktion der Trainingsdaten ist in linearer Laufzeit zu bewältigen. Weiterhin müssen zu diesen Trainingsdaten Record-Linkage-Durchläufe durchgeführt werden. Diese können je nach Größe der zugrunde liegenden Daten viel Zeit in Anspruch nehmen. Allerdings resultieren umfangreiche Record-Linkage-Durchläufe auch in umfangreichen Gewichtsdateien. Alternative unüberwachte Algorithmen haben eine kubische bzw. quadratische Laufzeit in Bezug auf die Anzahl der Gewichte innerhalb der Gewichtsdaten. Die Laufzeit solcher Algorithmen sollte also auf solch umfangreichen Gewichtsdateien sogar über der von CLARA liegen. Genauere Untersuchungen hierzu wären jedoch notwendig, um gültige Aussagen zu treffen.

CLARA übertraf auch die erreichte Klassifikationsgüte der manuellen Schrankenbestimmung anhand von Histogrammen. Dieses Ergebnis würde dafür sprechen, die manuelle Klassifikation komplett durch das CLARA-System zu ersetzen.

Da die beiden Systeme aber komplett unabhängig voneinander fungieren - CLARA basiert auf Trainingsdaten, manuelle Schrankenbestimmung auf Gewichtsdaten - bietet sich am ehesten eine Kombination der beiden Techniken an, bei der es also immer eine gegenseitige Kontrolle gäbe. Größere Abweichungen zwischen den Methoden würden also schnell Hinweis darauf geben, dass eine der Klassifikationsmethoden eine falsche Schranke vorhergesagt hat. Hierauf könnten gerade auf die manuelle Schrankensetzung Anpassungen folgen. An dieser Stelle mag es verwundern weshalb eine Kontrolle von CLARA überhaupt nötig ist, nachdem die F-Werte in den Ergebnissen so nah an den Optimalwerten liegen. Der Grund ist, dass überwachte Klassifizierung immer eine leichte Abweichung von einer optimalen Position haben wird. Bei Kenntnis des ungefähren Bereichs (gegeben durch überwachte Klassifizierung/CLARA) lässt sich die genaue Position manuell in ein lokales Minimum oder eine passende Lücke einpassen.

Laut Han et Al. gibt es zudem bei überwachten Klassifikationssystemen, wie z.B. CLARA, die Gefahr einer Überanpassung (Overfitting) der Trainingsdaten an die Testdaten, was sich negativ auf die Klassifikationsgüte auswirken könnte [41,80]. Diese Befürchtung war bei der Anwendung von CLARA nicht zu bestätigen. Wie sich anhand der Trainingsset-Varianten zeigte, war das Klassifikationsergebnis immer dann am höchsten, wenn die Verteilung der Ursprungswerte möglichst den Originalwerteverteilungen entsprach. Generell spielt Overfitting für das System keine Rolle da jeder Klassifikator immer für das gegebene Originaltestdatenset und nicht für andere Testdatensets einzeln generiert wird. Generell lagen die Klassifikationsergebnisse von CLARA unabhängig von der zugrunde liegenden Datenqualität der Testdatensätze extrem nah am erreichbaren Optimalwert.

4.5. Übertragung der Ergebnisse auf den aktuellen Stand der Wissenschaft

Die Klassifikationsergebnisse von CLARA zeigten auf einer umfangreichen Menge von Testdaten, dass überwachte Klassifikation, repräsentiert durch die CLARA-Technologie, unüberwachter Klassifikation, repräsentiert durch SLC und den SNN, grundsätzlich überlegen war. Eine Auswertung in solch einem Umfang, auf einer Menge von insgesamt 400 individuellen Testdatensätzen, hatte bisher noch nicht stattgefunden [38].

Manuelle Klassifikation, basierend auf Histogramm-Daten, schienbei guter Datenqualität valide und lag in dieser Arbeit konkret zwar unterhalb den Ergebnissen von CLARA, jedoch meist über den Ergebnissen der unüberwachten Technologie, jedoch ließ die Klassifikationsqualität auch hier bei schlechterer Datenqualität nach. Die überwachten Klassifikationssysteme sind hiervon unabhängig und sollten also gerade in Szenarien, in denen Datenqualitätsprobleme vorliegen, unterstützend genutzt werden. So würde sich zum Beispiel anbieten, eine Implementierung des CLARA-Systems auch in den kommenden Record-Linkage-Durchläufen der DKFS unterstützend einzusetzen. Da überwachte Systeme grundsätzlich etwas gröber klassifizieren (d.h. die vorhergesagte Schranke kann von der eigentlichen Position etwas abweichen) sollte jedoch eine Vollautomatisierung vermieden werden. Eine Kombination aus manueller und unterstützender Klassifikation scheint am wirkungsvollsten.

Neben dem Vergleich zwischen unüberwachter sowie überwachter Klassifikation wäre das Konzept zum CLARA-System an sich als weiterer Beitrag zum Stand der Wissenschaft zu nennen. Das CLARA System baut in dieser Arbeit grundsätzlich auf der Konstruktion von Trainingsdaten, anschließendem Record-Linkage auf diesen Daten, Bestimmung einer Schranke

auf den resultierenden Gewichtsdateien, sowie Einpassen der Schranke in das zugrunde liegende Testset auf. Da Projekte verschiedene Record-Linkage-Ansätze verwenden, sollten also die nicht zur Klassifikation gehörenden Schritte des Privacy-Preserving-Record-Linkage von CLARA entkoppelt werden. Würde man also eine Veröffentlichung von Software zu dieser Technologie anstreben, könnte man Tools zur Erzeugung von Trainingsdaten entsprechend der CLARA-Technologie sowie zur Ermittlung der Schranke auf den Gewichtsdateien der Trainingsdaten anbieten. Das System wäre dann mit jeder Art von auf Gewichten basierenden Record-Linkage-Systemen kompatibel. Für den User gäbe es lediglich zwei Parameter zu spezifizieren. Zum einen die Größe des Überlappungsbereiches, für den ein Empfehlungswert von 3% der Größe des kleineren Teilsets gegeben wird. Zum anderen ließe sich die Anzahl der Trainingssets spezifizieren, zu denen jeweils ein Klassifikator bestimmt wird, dessen Mittelwert den finalen Klassifikator darstellt (in dieser Arbeit etwa wurden zu jedem Testset jeweils 3 CLARA-Trainingssets erzeugt). Die Anwendung wäre also einfach handhabbar. Ein Kritikpunkt sowie eine Einschränkung wäre der zusätzlich benötigte Festplattenspeicherplatz, der durch die Erzeugung von Trainingsdaten freigehalten werden müsste.

4.6. **Limitierungen der Arbeit**

Nicht beantworten kann diese Arbeit, ob eventuell andere überwachte Klassifikationssysteme CLARA überlegen wären und wie gut CLARA hierbei vergleichsweise in Bezug auf die Klassifikationsgüte abschneiden würde. Alternative Konzepte wie Bumping, Bagging oder Multiview [83,84] oder die Verwendung von überwachten Regressionsbäumen klingen vielversprechend [100]. Vergleichende Arbeiten wären hierzu notwendig. Die Klassifikationsgüte von CLARA erschien jedoch in der vergleichenden Analyse, basierend auf den maximal möglichen F-Werten bereits so gut, dass der Methodik eventuell aufgrund der einfachen Anwendbarkeit der Vorzug vor anderen Methoden gegeben werden sollte. Innovativ ist auch die absolute Unabhängigkeit von Trainingsdaten, da diese komplett aus den Originaldaten generiert werden, sowie die eindeutige Konfiguration, die in anderen Arbeiten nicht in dieser Art spezifiziert wurde, wodurch Unklarheiten in der Anwendung vermieden werden. Eine Vollautomatisierung der Klassifikation wäre damit unabhängig von den Testdaten problemlos möglich.

Trotz der auf den Testdaten gegebenen guten Abgleichsgüte gibt es Sonderfälle, mit denen das System nicht gut umgehen kann und die auch hier zu einer starken Fehlklassifikation führen können. Würden etwa per Zufall ausschließlich Links mit einem extrem hohen Abgleichsgewicht (beispielsweise bei doppelten Vornamen) dem Überlappungsbereich

hinzugefügt werden, würde ein darauf resultierender Klassifikator alle echten Übereinstimmungen, unterhalb dieser Links als falsch klassifizieren. Der Lösungsansatz um unglückliche Zufallsziehungen zu umgehen, ist die Erzeugung mehrerer Klassifikatoren und hierbei die Wahl des Median bzw. des Mittelwertes der vorhergesagten Schrankenwerte. In den Analysen dieser Arbeit wurden hierfür jeweils drei CLARA-Trianingssets konstruiert. Je nach Leistungskraft der zugrunde liegenden Hardware und Umfang der angestrebten Arbeiten könnten aber weitere Trainingsdaten das Risiko einer starken Fehlklassifikation verringern.

Grundsätzlich handelt es sich bei CLARA außerdem nicht formell um eine überwachte Klassifikation, sondern eher um eine semi-überwachte Klassifikation, da echte Übereinstimmungen, die jedoch nicht bekannt sind, das Ergebnis der vorhergesagten Klassifikatoren eventuell negativ beeinflussen können. Basierend auf den guten Ergebnissen erscheint dieser Einfluss aber nicht mit allzu großen negativen Konsequenzen einherzugehen.

Weitere Einschränkungen wie Laufzeit oder auch benötigter Festplattenspeicher wurden bereits angesprochen, erscheinen jedoch für die meisten Projekte als eher unproblematisch.

Weiterhin wäre zu erwähnen, dass den Analysen in dieser Arbeit stets ein probabilistisches Record-Linkage-System zu Grunde lag. Bei der Gewichtsberechnung spielen hierbei auch Häufigkeiten und dementsprechend Werteverteilungen eine große Rolle. Das CLARA-System wurde entsprechend für Variationen von Trainingssets, die eben genau in diesen Werten variieren, konzipiert. Für das probabilistische Record-Linkage bewährte sich dies als nachvollziehbarer Ansatz. Approximatives Record-Linkage jedoch, bei dem es sich aller Voraussicht nach um die Zukunftstechnologie im Bereich des Privacy-Preserving-Record-Linakge handelt, ist von Häufigkeiten zum jetzigen Stand der Wissenschaft, soweit dem Autor dieser Arbeit bekannt, unabhängig. Dennoch wäre anzunehmen, dass das CLARA-System auch auf approximatives Record-Linkage anwendbar wäre unter der Prämisse Fehler bei der Konstruktion von Trainignsdaten zu Berücksichtigen. Ohne Berücksichtigung der Fehlerhäufigkeiten würden hier semtliche Abgleiche im Überlappungsbereich in einem Wert von 1.0 resultieren. Hierbei wären jedoch möglicherweise Laufzeitoptimierungen, zum Beispiel, eine Verkleinerung der Trainingssets oder Ähnliches denkbar. Das approximative Record-Linkage sollte grundsätzlich weniger von der Parametrisierung der Trainingssets beeinflusst werden. Um Eindeutigkeit zu bewahren, wäre der CLARA-Ansatz aber auch hier sicherlich einsetzbar. Eine geprüfte Empfehlung kann jedoch im Moment nur für den Einsatz auf probabilistischen Record-Linkage-Systemen gegeben werden.

5. Zusammenfassung

Im Zuge einer Studie zu familiärem Darmkrebs wurde ein probabilistisches Privacy-Preserving-Record-Linkage umgesetzt, dass den anonymen Abgleich zwischen Studienteilnehmern und eingetragenen Patienten des Münchner Tumorregisters erlaubte. Bei dieser Aufgabe konnten Probleme im Bereich der Klassifikation identifiziert werden. Um die hierbei verwendete manuelle Klassifikation zu unterstützen, wurde nach alternativen, binären Klassifikationssystemen gesucht. Die existierenden Techniken gingen jedoch meist mit neuen Unsicherheitsfaktoren einher und es fehlte an umfangreichen Vergleichen und erfolgreichen Einsatzberichten. Ziel dieser Arbeit war es daher, eine leicht einsetzbare Klassifikationstechnik zu konzipieren, die bei der manuellen Klassifikation unterstützend eingesetzt werden konnte und dabei anderen Methoden in der Klassifikationsgüte überlegen war.

Bei der neu konzipierten Technik handelte es sich um ein überwachtes Klassifizierungssystem, das die Klassifikatoren anhand von künstlichen Trainingsdaten, die direkt aus den zu vergleichenden Daten generiert wurden, vorhersagte. Entsprechend der Beschreibung wurde das System CLARA benannt (**CLA**ssification for **R**ecord-Linkage with **A**rtificial Trainingssets). Die genaue Parametrisierung zur Erzeugung dieser Trainingsdaten wurde über Analysen zu Variationen in den genannten Trainingsdaten optimiert.

Das System wurde gegenüber Techniken aus dem Bereich der unüberwachten Klassifikation getestet. Der Test enthielt auch einen Vergleich zur manuellen Schrankensetzung. Testgrundlage waren 400 auf klinischen Realdaten basierende Testsets, die sich jeweils in mindestens einem der Parameter Größe, Überlappung bzw. Datenqualität unterschieden. Anhand der vergleichenden Analyse ergab sich, dass das CLARA System den anderen Techniken stark überlegen war. Besonders auf Ausgangsdaten mit problematischer Datenqualität hielt CLARA die hohe Klassifikationsqualität, also in Szenarien, in denen unüberwachte Klassifikationen und auch manuelle Klassifikation oft mit Problemen behaftet sind. Ein weiteres Merkmal von CLARA war die einfache Anwendung, bei der es kaum zu Unsicherheiten kommen konnte. Eine öffentlich zugängliche Implementierung des Systems wurde noch nicht erstellt, ist aber für die nahe Zukunft geplant.

Letztendlich lieferten die Analysen Indiz für die Überlegenheit der überwachten Klassifikationssysteme gegenüber den unüberwachten Klassifikationssystemen im Bereich des Record-Linkage. Überwachte Systeme bieten zudem eine von der manuellen Schrankensetzung unabhängige Sichtweise, weswegen diese sehr gut in Kombination verwendet werden könnten.

6. Literaturverzeichnis

1. *Third Quarter 2013 Financial Summary*. California: Facebook, Inc.; Oct., 2013.
2. *Google – Privacy Policy.* Available from: http://www.google.de/policies/privacy/
3. Braun S, Flaherty A, Gillum J, Apuzzo, M. *Secret to PRISM Program: Even Bigger Data Seizures.* Associated Press; 2013.
4. Kramer M. *The NSA Data: Where Does It Go?.* National Geographic – Daily news; 2013.
5. Hauf D. *Allgemeine Konzepte: K-Anonymity, l-Diversity and T-Closeness*. IPD Uni-Karlsruhe; 2008.
6. Pommerening K. *Datenschutz in medizinischen Informationssystemen*. MedReport. 1995; 9(19):6-7.
7. Meisinger C, Löwel H, Mraz W, König W. *Prognostic value of apolipoprotein B and A-I in the prediction of myocardial infarction in middle-aged men and women: results from the MONICA/KORA Augsburg cohort study*. Eur Heart J. 2005; 26: 1–8.
8. Steinke C. *Deutschlands größte Gesundheitsstudie geht in die zweite Runde*. Pressemitteilung der Universität Greifswald beim Informationsdienstes Wissenschaft. 2012
9. Bundeskrebsregisterdatengesetz vom 10. August 2009 (BGBl. I S. 2707)
10. Li N, Li T, Venkatasubramanian *S. T-Closeness: Privacy Beyond k-Anonymity and l-Diversit*y. Data Engineering. 2007.
11. *HIPAA Administrative Simplification*. U.S. Department of Health and Human Services Office for Civil Rights. 2013.
12. Sweeney L. *K-anonymity: A model for protecting privacy*. International journal of uncertainity, fuzziness and knowledge-based systems.2002; 10(5):557 – 570.
13. Dunn H. *Record Linkage*. American Journal of Public Health. 1946;36(12):1412.
14. Schnell R, Bachteler T, Reiher J. *Privacy-preserving record linkage using Bloom filters*. BMC Medical Informatics and Decision Making. 2009 Aug 25;9:41.
15. V, Karakasidis A, Mitrogiannis V. *Privacy Preserving Record Linkage approaches*. Int. J. of Data Mining, Modelling and Management. 2009;1:206-221.
16. Trepetin S. *Privacy-preserving string comparisions in record linkage systems: a review*. Information Security Journal: A Global Perspective. 2008; 17:253-266.
17. Karakasidis A, Verykios V. E-Activity and Intelligent Web Construction; Idea Group Reference. 2011. *Advances in privacy preserving record linkage*.
18. Durham E, Kantarcioglu M, Malin B. *Quantifying the correctness, computational complexity, and security of privacy-preserving string comparators for record linkage*. Inf Fusion. 2012 Oct 1;13(4):245-259.
19. Quantin C, Bouzelat H, Allaert FA, Benhamiche AM, Faivre J, Dusserre L. *Automatic record hash coding and linkage for epidemiological follow-up data confidentiality*. Methods Inf Med. 1998 Sep;37(3):271-277.
20. Quantin C, Bouzelat H, Allaert FA, Benhamiche AM, Faivre J, Dusserre L. *How to ensure data security of an epidemiological follow-up: quality assessment of an anonymous record linkage procedure*. Int J Med Inform. 1998 Mar;49(1):117-122.

21. Mansmann U, Stausberg J, Engel J, Heussner P, Birkner B, Maar C. *Familien schützen und stärken – Umgang mit familiärem Darmkrebs. Eine Pilotstudie zur Inzidenz von Risikoclustern und zur Möglichkeit ihrer Detektion.* Der Gastroenterologe 2012; 7: 271-272.
22. Jemal A, Bray F, Center MM, Ferlay J, Ward E, Forman D. *Global cancer statistics.* CA Cancer J Clin. 2011 Mar-Apr;61(2):69-90.
23. Watson AJ, Collins PD. *Colon cancer: a civilization disorder. Digestive diseases*. 2011;29(2):222-8.
24. Schneider R. *Das Lynch-Syndrom – Epidemiologie, Klinik, Genetik, Screening, Therapie*. Zeitschrift für Gastroenterologie. 2012; 50: 217-225
25. Fotiadis C, Tsekouras DK, Antonakis P, Sfiniadakis J, Genetzakis M, Zografos GC. *Gardner's syndrome: a case report and review of the literature.* World J Gastroenterol. 2005 Sep 14;11(34):5408-5411.
26. Slattery ML, Levin TR, Ma K, Goldgar D, Holubkov R, Edwards S. *Family history and colorectal cancer: predictors of risk.* Cancer Causes Control. 2003 Nov;14(9):879-887.
27. Jeffery GM1, Hickey BE, Hider P. *Follow-up strategies for patients treated for non-metastatic colorectal cancer.* Cochrane Database Syst Rev. 2002;(1)
28. Cunningham D, Atkin W, Lenz HJ, Lynch HT, Minsky B, Nordlinger B, Starling N.*Colorectal Cancer*. Lancet. 2010 Mar 20;375(9719):1030-1047.
29. He J, Efron JE. *Screening for colorectal cancer*. Adv Surg. 2011;45:31-44.
30. Hewitson P1, Glasziou P, Watson E, Towler B, Irwig L.. *Cochrane systematic review of colorectal cancer screening using the fecal occult blood test (hemoccult): an update.* Am J Gastroenterol. 2008 Jun;103(6):1541-1549.
31. Fellegi I, Sunter A. A Theory for Record Linkage. Journal of the American Statistical Association. 1969; 64 (328): 1183–1210.
32. Jaro M. *Probabilistic linkage of large public health data files*. Stat Med. 1995 Mar 15-Apr 15;14(5-7):491-498.
33. Blakely T, Salmond C. *Probabilistic record linkage and a method to calculate the positive predictive value*. International Journal of Epidemiology. 2002 Dec; 31(6):1246-1252.
34. Nasseh D, Engel J,Mansmann U, Tretter W, Stausberg J. *Matching study to registry data: maintaining data privacy in a study on family based colorectal cancer.* Fullpaper accepted for MIE 2014.
35. Pommerening K, Drepper J, Ganslandt T, Helbing K, Müller T, Sax U, Semler S, Speer R. *Das TMF-Datenschutzkonzept für medizinische Daten-sammlungen und Biobanken*. Proceeding of: Informatik 2009: Im Focus das Leben, Beiträge der 39. Jahrestagung der Gesellschaft für Informatik e.V. (GI); 2009; Lübeck.
36. Daemen J, Rijmen V. *AES Proposal:* Rijndael; 1999.
37. Palanisamy V, Jeneba M. *Hybrid cryptography by the implementation of RSA and AES. International Journal of Current Research*. April 2011;33(4): 241-244
38. Christen P. *Data Matching: Concepts and Techniques for Record Linkage, Entity Resolution, and Duplicate Detection*. Berlin Heidelberg: Springer; 2012.

39. Meyer M. *Kontrollnummern und Record Linkage*. Das Manual der epidemiologischen Krebsregistrierung. Hentschel S, Katalinie A, editor. Zuckschwerdt; 2011:57-68.
40. Kieschke J. *Methoden von Registern für die Versorgungsforschung.*DNVF-Springschool. 2013.
41. Han J, Kamber M. *Data Mining: concepts and techniques*. 2nd edition. San Francisco: Morgan Kaufmann; 2006.
42. Mitchell T. *Machine Learning*. USA: McGraw Hill; 1997.
43. Gill L. *Methods for automatic record matching and linking and their use in national statistics*. Tech. Rep. Methodolgy Series no. 25; 2001.
44. Newcombe HB. *Handbook of Record Linkage: Methods for Health and Statistical Studies, Administration, and Business. Oxford University Press, Inc*. 1988; New York.
45. Jonas J, Harper J. *Effective counterterrorism and the limited role of predictive data mining*. Policy Analysis. 2006; 584.
46. Manghi P. Mikulicic M. *PACE: A general-purpose tool for authority control*. Metadata and Semantic Research; 2011. 80-92.
47. Fogel R. *New sources and new techniques for the study of secular trends in nutritional status, health, mortality, and the process of aging*. NBER Historical Papers. 1993.
48. Glasson E, De Klerk N, Bass A, Rosman D, Palmer L, Holman C. *Cohort profile: the Western Australian family connections genealogical project*. International Journal of epidemiology. 2008 Feb;37(1):30-35.
49. Newcombe H, Kennedy J. *Record linkage: making maximum use oft he disciminating power of identifying information*. Communications of the ACM. 1962;5(11):31-88.
50. Newcombe H, Kennedy J, Axford S, James A. *Automatic linkage of vital records*. Science. 1959; 130(3381):954-959.
51. Winkler WE, Thibaudeau Y. *An application of the Fellegi-Sunter model of record linkage to the 1990 U.S. decennial census*. Tech. Rep. RR1991/09. 1991.
52. Bloom B. *Space/time trade offs in hash coding with allowable errors*. Communictation of the ACM. 1970; 13(7):422-426.
53. Durham E,Xue Y, Kantarcioglu M, Malin B. *Private Medical Record Linkage with Approximate Matching*. AMIA Annu Symp Proc. 2010; 2010: 182–186.
54. Churches T, Christen P: *Some methods for blindfolded record linkage*. BMC Med Inf Decis Mak 2004; 4(9).
55. Hinrichs H. *Bundesweite Einführung eines einheitlichen Record Linkage Verfahrens in den Krebs-registern der Bundesländer nach dem KRG, Abschlussbericht, Projekt Deutsche Krebshilfe*. Antragsnummer 70-2043-Ap I. OFFIS. Oldenburg; 1999
56. Russell RC. *SOUNDEX (untitle*d). US patent 1261167. 1918.
57. Postel H.-J. *Die Kölner Phonetik – Ein Verfahren zur Identifizierung von Personennamen auf der Grundlage der Gestaltanalyse*. IBM-Nachrichten. 19 (1969); 925-931.

58. Appelrath HJ, Michaelis J, Schmidtmann I, Thoben W. *Empfehlung an die Bundesländer zur technischen Umsetzung der Verfahrensweisen Gemäß Gesetz über Krebsregister (KRG*). Informatik, Biometrie und Epidemiologie in Medizin und Biologie 1996;27: 101-110.
59. Krieg V, Hense HW, Lehnert M, Mattauch V. *Record Linkage mit kryptographierten Identitätsdaten in einem bevölkerungsbezogenen Krebsregister*. Entwickung, Umsetzung und Fehlerraten. Gesundheitswesen. 2001; 63: 376-382.
60. Thoben W, Apelrath H.-J, Sauer S. *Record Linkage of Anonymous Data by Control Numbers*. In: W.Gaul, D.Pfeifer. From Data to Knowledge: Theoretical and Practical Aspects of Classification, Data Analysis and Knowledge Organisation. Springer ; 1994: 412-419.
61. Floyd J. *What do Hash Collisions Really Mean?* Available at: http://permabit.wordpress.com/: Permabits and Petabytes. 2011.
62. Gilbert H, Handschuh H. *Security Analysis of SHA-256 and Sisters*. Selected Areas in Cryptography. 2003; 175–193
63. Stevens M. *Cryptanalysis of MD5 & SHA-1*. Available at: http://2012.sharcs.org/slides/stevens.pdf. 2012.
64. Krawczyk H, Bellare M, Canetti R. *HMAC: Keyed-Hashing for Message Authentication*. RMC 2014.
65. Morris R, Thompson K. *Password Security: A Case History*. Bell Laboratories. 1978.
66. Kirsch A, Mitzenmacher M. *Less hashing, same performance: building a better Bloom filter.* Algorithms-ESA 2006. Proceedings of the 14th Annual European Symposium; September 2006; 11-13.
67. Nasseh D, Stausberg J. Impact of variations in Anonymous Record Linkage on Weight Distribution and Classification. Stud Health Technol Inform. 2013;192:922.
68. Hernandez MA, Stolfo SJ. *The merge/purge problem for large databases*. ACM SIGMOND. 1995; 127-138.
69. Hernandez MA, Stolfo SJ. *Real-world data is dirty. Data cleansing and the merge/purge problem*. Data Mining and Knowledge Discovery. 1998; 2(1):9-37.
70. Christen P. *A Survey of Indexing Techniques for Scalable Record Linkage and Deduplication*. Knowledge and Data Engineerin; 24(9).
71. Christen P. *Automatic Record Linkage using Seeded Nearest Neighbour and Support Vector Machine Classification.* KDD '08 Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining. New York, 2008; 151-159.
72. Pommerening K, Sariyar M. Der PID-Generator der TMF. TMF-Workshop „Tools zum ID-Management in der klinischen Forschung“. 2010.
73. Schmidtmann I, Hammer G, Sariyar M, Gerhold-Ay A. Evaluation des Krebsregisters NRW – Schwerpunkt Record Linkage. Final report 11 Jun 2009. Mainz (DE): Universitätsmedizin der Johannes Gutenberg-Universität Mainz, Körperschaft des öffentlichen Rechts; 2009. 50.
74. Winkler WE. *Using the EM Algorithm for Weight Computation in the Fellegi-Sunter Model of Record Linkage*. Proceedings of the Section on Survey Research Methods, American Statistical Association. 2000.

75. Silveira D, Artmann E. Accuracy of probabilistic record linkage applied to health databases: systematic review. Rev Saúde Pública 2009.
76. Dice LR. *Measures of the amount of ecologic association between species*. Ecology. 1945; 26(3):297-302.
77. Sibson R. *SLINK: an optimally efficient algorithm for the single-link cluster method*. The Computer Journal (British Computer Society). 1973; 16(1): 30–34.
78. Defays D. (1977). *An efficient algorithm for a complete link method*. The Computer Journal (British Computer Society). 1977; 20 (4): 364–366.
79. Sarawagi S, Bhamidipaty A. *Interactive deduplication using active learning*. ACM KDD'02. 2002: 269–278.
80. Mitchell TM. *Machine Learning*. McGraw Hill. 1997.
81. Breimann L, Freidman J, Olshen R, Stone C. *Classification and regression trees*. Chapman and Hall/CRC. 1984.
82. Christen P. *Febrl - a freely available record linkage system with a graphical user interface*. HDKM'08, CRPIT vol. 80. 2008.
83. Sariyar M, Borg A. *Bagging, bumping, multiview, and active learning for record linkage with empirical results on patient identity data*. Comput Methods Programs Biomed. 2012 Dec;108(3):1160-1169.
84. Sariyar M, Borg A, Pommerening K. *Evaluation of Record Linkage - Methods for Iterative Insertion.* Methods Inf Med. 2009;48(5):429-437
85. Yancey WE. *Big Match – A program for extracting probable matches from a large file for record linkage*. Tech Rep RRC2007/01. 2007.
86. Bilgic M, Licamele L, Getoor L, Shneiderman B. *D-Duple: An interactive tool for entity resolution in social networks*. IEEE Symposium on Visual Analytics, Science and Technology. 2006: 43-50.
87. Draisbach U, Naumann F. *Dude: The duplicate detection toolkit*. Workshop on Quality in Databases. 2010.
88. Jurczyk P, Lu J, Xiong L, Cragan J, Correa A. *FRIL: A tool for comparative record linkage. AMIA Annual Symposium Proceedings*. 2008: 440.
89. Schnell R, Bachteler T, Bender S. *A toolbox for record linkage. Austrian Journal of Statistics.* 2004; 33(1&2):125-133.
90. Talburt J. *Entity Resolution and Information Quality*. Morgan Kaufmann. 2011.
91. Sariyar M. Borg A. *The Record Linkage package*. Detecting errors in data. The R Journal. 2010; 2(2):61-67.
92. Jentzsch A, Isele R, Bizer C. *Silk-generating RDF links while publishing or consuming linked data*. Poster at the International Semantic Web Conference. 2010.
93. Elfeky MG, Verykios V, Elmagarmid AK. *TAILOR: A record linkage toolbox*. IEEE ICDE. 2002: 17-28.
94. Cohen W. *The WHIRL approach to data integration. IEEE Intelligent Systems*. 1998; 13(3):20-24.
95. Christen P, Verykios V, Vatsalan D. *A Tutorial on Techniques for Scalable Privacy-preserving Record Linkage.* CIKM 2013.

96. Powers D. *Evaluation: From Precision, Recall and F-Measure to ROC, Informedness, Markedness & Correlation.* Journal of Machine Learning Technologies. 2011; 2 (1): 37–63.
97. Versi E. *"Gold standard" is an appropriate term*. BMJ. 1992.
98. Heller B. *Fragen der Philosophie 1: Zugänge*. Books on Demand GmbH. 2000
99. Kaufman L. Rousseeuw PJ. *Finding Groups in Data: An Introduction to Cluster Analysis*. John Wiley. 1990.
100. Therneau TM, Atkinson EJ. *An Introduction to Recursive Partitioning Using the Rpart Routine*. Mayo Clinic, Section of Biostatistics, Rochester. 1997.
101. Boonchai K, Speedie S, Connelly D. *Linking patients' records across organizations while maintaining anonymity*. AMIA 2007 Symposium Proceedings Page. 2007: p.1008.
102. Contiero P, Tittarelli A, Tagliabue G, Maghini A, Fabiano S, Crosignani P, Tessandori R. *The EpiLink Record Linkage Software -Presentation and Results of Linkage Test on Cancer Registry Files*. Methods Inf Med 1. 2005.
103. Fonseca M, Coeli C, Lucena F, Veloso V, Carvalho M. *Accuracy of a probabilistic record linkage strategy applied to identify deaths among cases reported to the Brazilian AIDS surveillance database.* 2010.
104. Migowski A, Chaves RB, Coeli CM, Ribeiro AL, Tura BR, Kuschnir MC, Azevedo VM, Floriano DB, Magalhães CA, Pinheiro MC, Xavier RM. *Accuracy of probabilistic record linkage in the assessment of high-complexity cardiology procedures*. Rev Saude Publica. 2011 Apr;45(2):269-75.
105. Fournel I, Schwarzinger M, Binquet C, Benzenine E, Hill C, Quantin C. *Contribution of Record Linkage to Vital Status Determination in Cancer Patients*. Stud Health Technol Inform. 2009;150:91-95.

7. Anhang

A. Abkürzungsverzeichnis

AES	**A**dvanced **E**ncryption **S**tandard
BDSG	**B**undes**d**aten**s**chutz**g**esetz
CLARA	**CLA**ssification for **R**ecord-Linkage with **A**rtificial Trainingssets.
CLINK	Bezeichnung eines effizienten Complete-Linkage-Clustering Ansatzes
DKFS	Studie zu familiärem Darmkrebs
DuDe	The **Du**plicate **De**tection Toolkit
FEBRL	**F**reely **E**xtensible **B**iomedical **R**ecord **L**inkage
FM	**F-M**easure
FN	**F**alse-Negatives (Falsch Negative)
FP	**F**alse-**P**ositives (Falsch Positive)
FRIL	**F**ine-Grained **R**ecords **I**ntegration and **L**inkage
HMAC	**H**ash-based **m**essage **a**uthentication **c**ode
IDAT	**I**dentifizierende **Dat**en
KNN	**K**-**N**earest-**N**eighbour
KORA	**KO**operative Gesundheitsforschung in der **R**egion **A**ugsburg
MDAT	**M**edizinische **Dat**en
MD5	**M**essage-**D**igest-Algorithmus (Version 5)
NSA	**N**ational **S**ecurity **A**gency
OYSTER	**O**pen s**YST**em **E**ntity **R**esolution
PPV	**P**ositive-**P**redictive-**V**alue (Positiver prädiktiver Wert)
PRISM	**P**lanning Tool for **R**esource **I**ntegration, **S**ynchronization and **M**anagement
RSA	**R**ivest, **S**hamir und **A**dleman (Initialen der Entwickler)
SHA	**S**ecure **H**ash **A**lgorithm
SLC	**S**ingle-**L**inkage-**C**lustering
SLINK	Bezeichnung eines effizienten Single-Linkage-Clustering Ansatzes
SNN	**S**eeded-**N**earest-**N**eighbour
SVM	**S**upport-**V**ector-**M**aschine
TAILOR	**R**ec**O**rd **LI**nk**A**ge **T**oolbox (Acronym rückwärts)
TMF	**T**echnologie- und Methodenplattform für die vernetzte **m**edizinische **F**orschung
TN	**T**rue-**N**egatives (Echt Negative)
TP	**T**rue-**P**ositives (Echt Positive
TRM	**T**umor**r**egister-**M**ünchen
UNICON	**Un**iform **Co**ntrol **N**umber Generator
WHIRL	**W**ord-Based **H**eterogeneous **I**nformation **R**epresentation Language

B. Tabellenverzeichnis

C. Formelverzeichnis

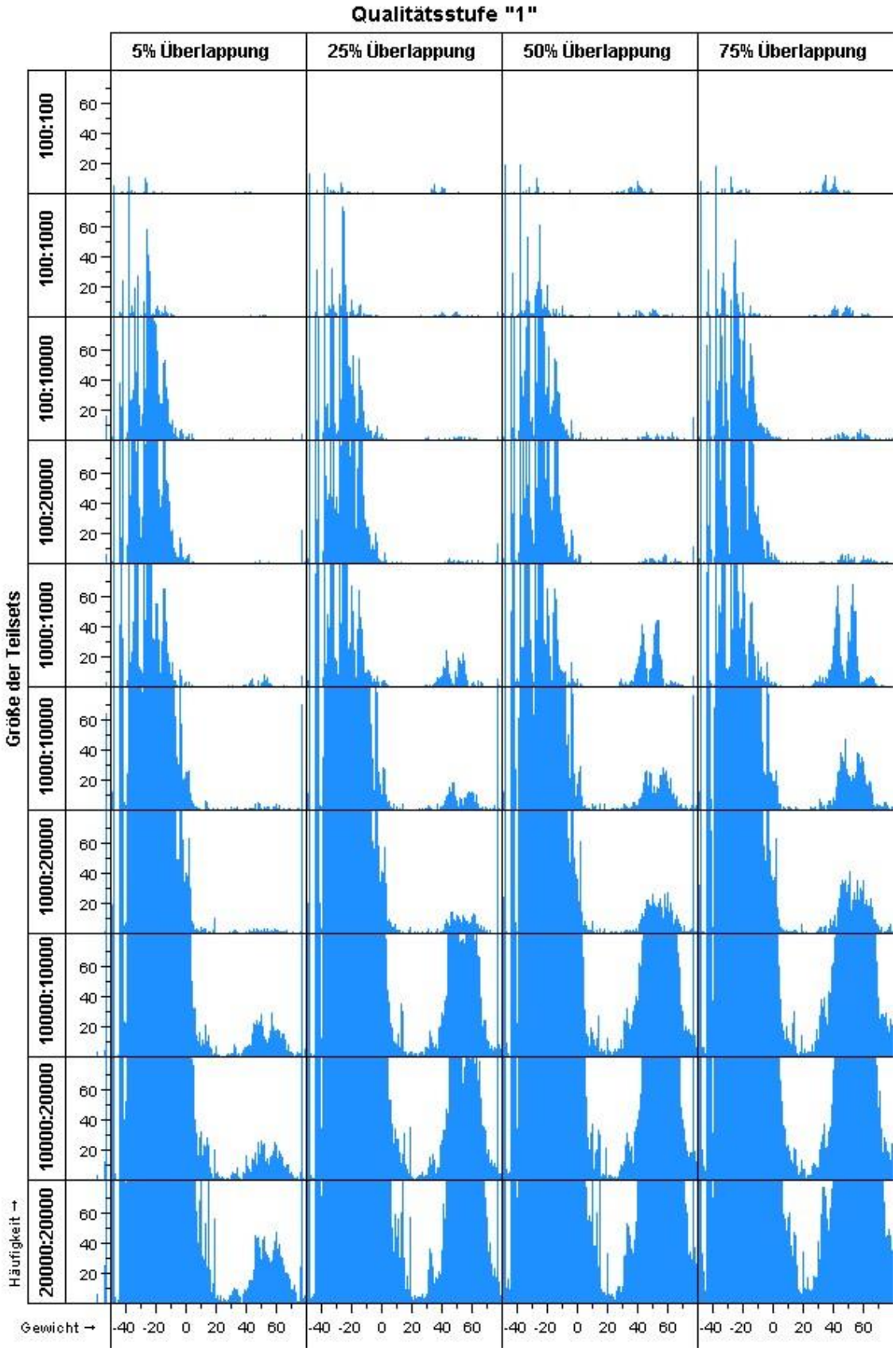
D. Abbildungsverzeichnis

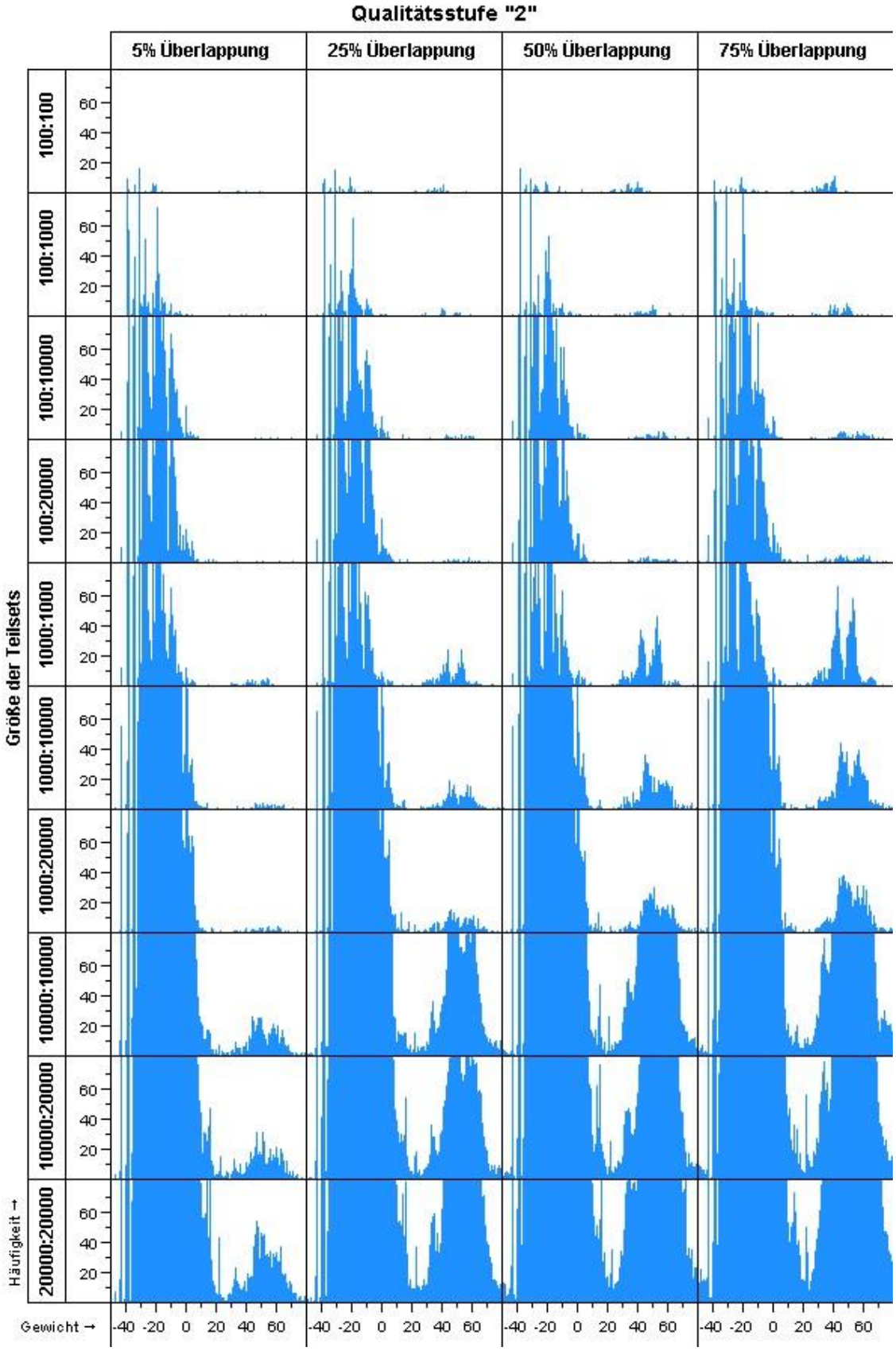
E. Programmverzeichnis

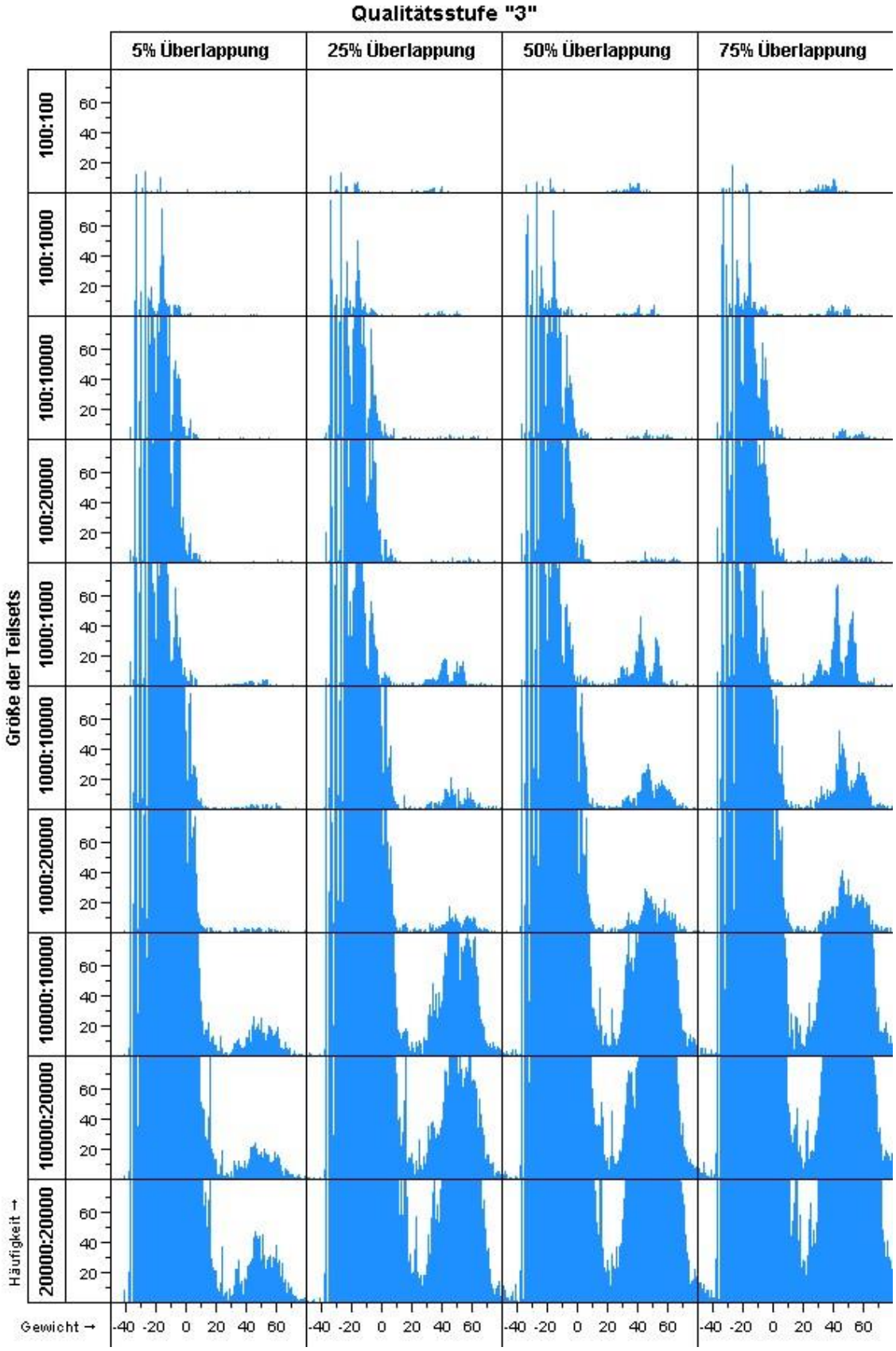
Index	Programmname	Funktion (Kurzbeschreibung)	Seite
1	*RecordLinkage*	Hauptklasse zur Durchführung eines Record Linkage auf zwei gegebenen Datensets.	37
2	*RecordLinkageInput*	Regelt das Einlesen der Daten für *RecordLinkage<1>*.	37
3	*Person*	Zu *RecordLinkageInput<2>* assoziierte Klasse.	-
4	*ConfigReader*	Zu *Record Linkage<1>* assoziierte Klasse.	-
5	*ListComparator*	Zu *RecordLinkage<1>* assoziierte Klasse.	-
6	*GenerateControlnumbers*	Klasse zur Standardisierung und Einwegverschlüsselung identifizierender Daten.	37
7	*CreateTestsets*	Klasse zur Erzeugung der 400 in dieser Arbeit verwendeten Testdatensätze.	44
8	*CreateTestSetsWeights*	Automatisierter Aufruf der Klasse Record Linkage auf den 400 gegebenen Testsets.	52
9	*FMeasure*	Berechnung des maximalen FMeasures auf den 400 Gewichtsdateien der Testsets.	52
10	*CreateTemplateTrainingsset*	Erzeugung eines Trainingssets unter Verwendung der Konstruktionsparameter eines zugrunde liegenden Testsets.	57
11	*CreateTrainingSetWeights*	Erzeugt zu semtlichen Trainingssets die Gewichtsdateien.	57
12	*MassFMeasure*	Erzeugt zu den Gewichtsdateien von Trainingssets die FMeasure und Schrankenwerte.	57
13	*FitBorderToTestset*	Fügt einen vorhergesagten Klassifikator in ein Testset ein und bemisst den hierdurch erzielten F-Measure-Wert.	58
14	*CreateSizeVariant1Trainingsset*	Erzeugung von Trainingssets deren Größe auf 100 festgelegt wurde.	59
15	*CreateSizeVariant1Trainingsset*	Erzeugung von Trainingssets deren Größe im Vergleich zu den Testdaten halbiert wurde.	60
16	*CreateErrorVariantTrainingsset*	Erzeugung von Trainingssets ohne Fehler im Überlappungsbereich.	60
17	*CreateOverlapVariant1Trainingsset*	Erzeugung von Trainingssets deren Überlappungsbereich auf 90% der Größe des kleineren Teilsets festgelegt wurde.	61
18	*CreateOverlapVariant2Trainingsset*	Erzeugung von Trainingssets deren Überlappungsbereich auf 30% der Größe des kleineren Teilsets festgelegt wurde.	61
19	*CreateOverlapVariant3Trainingsset*	Erzeugung von Trainingssets deren Überlappungsbereich auf 3% der Größe des kleineren Teilsets festgelegt wurde.	61
20	*CreateDistributionVariant1Trainingsset*	Erzeugung von Trainingssets in denen die Werteverteilungen der Patienten gleichverteilt wurden.	61
21	*AutomateTrainingsetProduction*	Klasse die die Produktion der 9600 Trainingssetvarianten automatisiert.	63
22	*CreateFinalTrainingsset*	Trainingsseterzeugung entsprechend dem CLARA Konzept.	63
23	*SingleLinkageNAIV*	Vereinfachung des Single Linkage Clusterings. Da es sich bei Gewichtsdateien um eindimensionale Daten handelt ist der Algorithmus trivial und bestimmt die größten Abstände in den Gewichtsdateien als Schrankenwert.	64
24	*KNN_Seed1*	Neares-Neighbour-Algorithmus mit k = 3 und Seedmenge nach Formel 17 und negativem Seetanteil von 5% bestimmt.	66
25	*KNN_Seed2*	Neares-Neighbour-Algorithmus. Die Seedmengen wurden per Treshhold festgelegt. Oberer Schrankenwert liegt hierbei bei +45 unterer Schrankenwert bei -15.	66
26	*CreateHistogramms*	Erzeugung von 400 Histogrammen zu den Testsets .	67

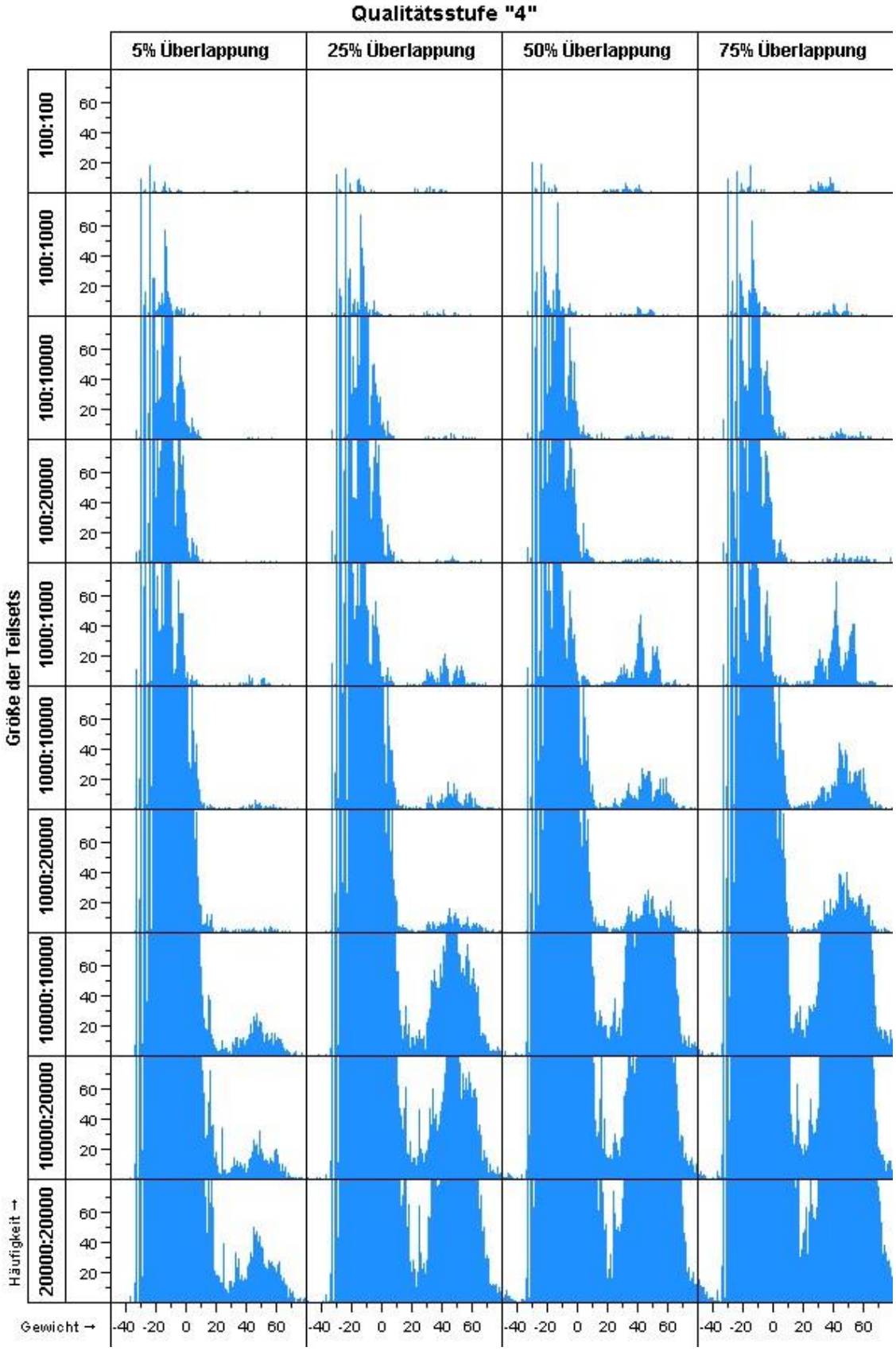
Einsicht in den Quellcode der Programme kann beim Autor dieser Arbeit direkt beantrag werden.

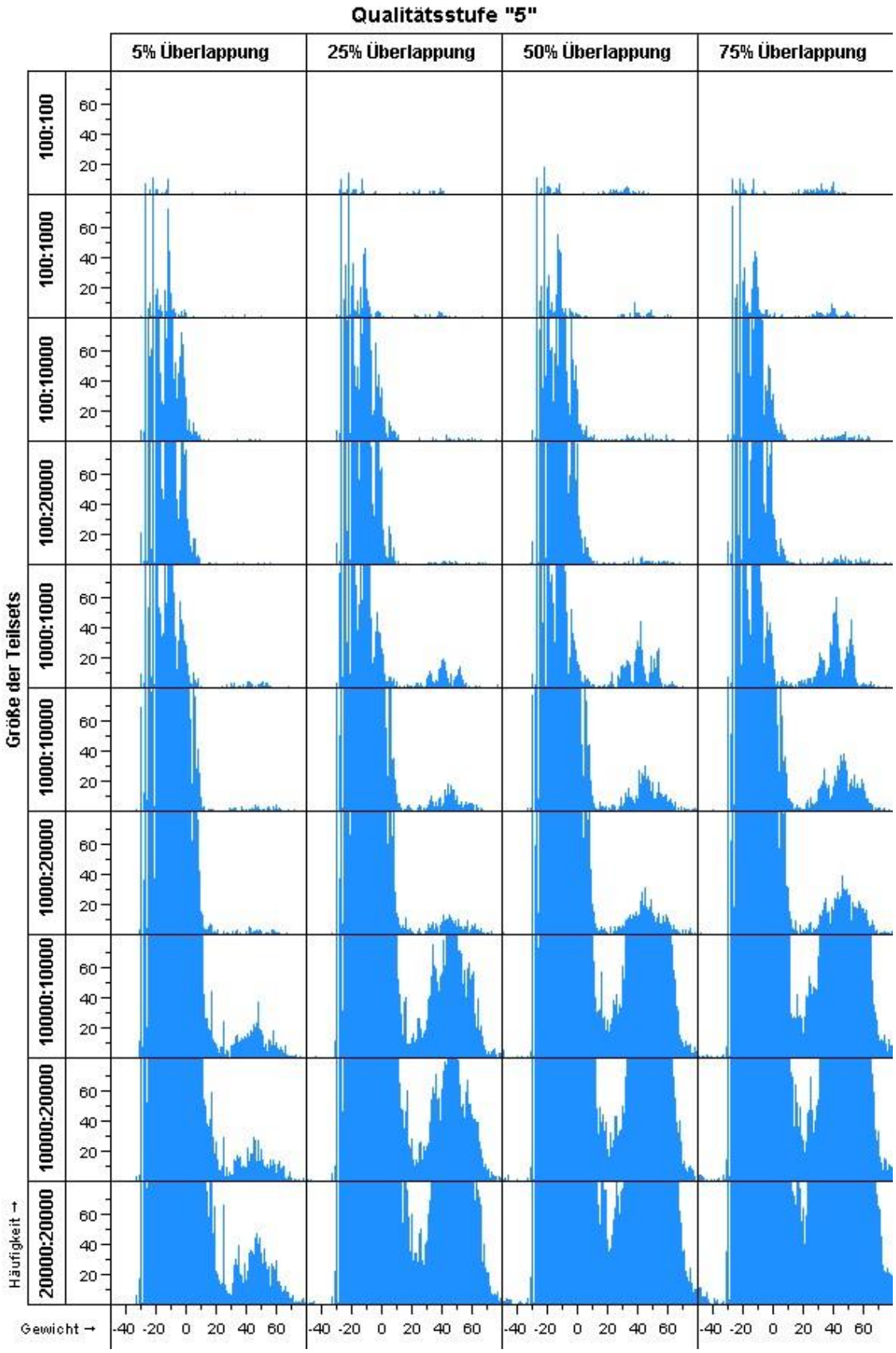
F. Histogramm-Übersicht der Testdatensätze

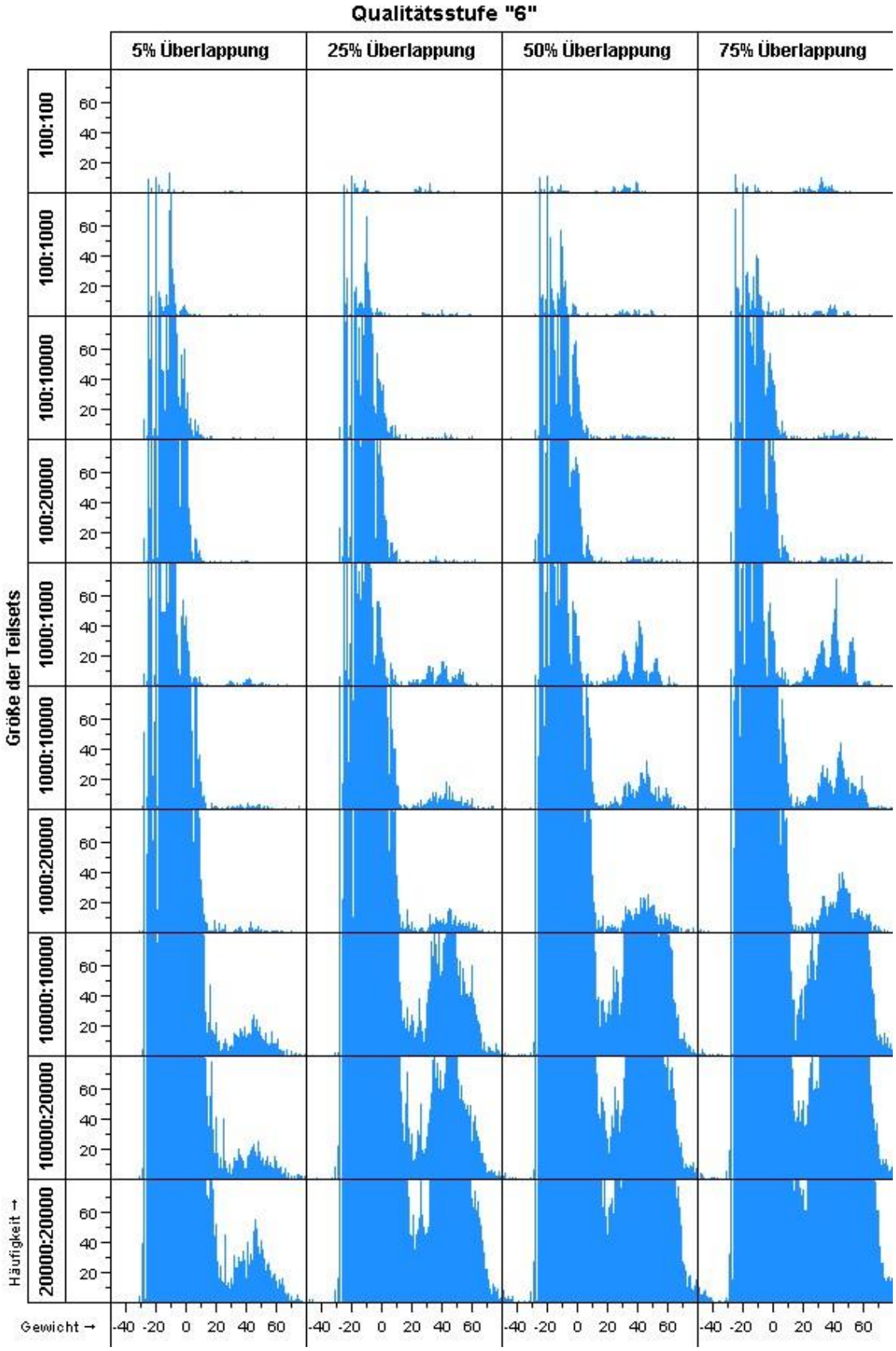


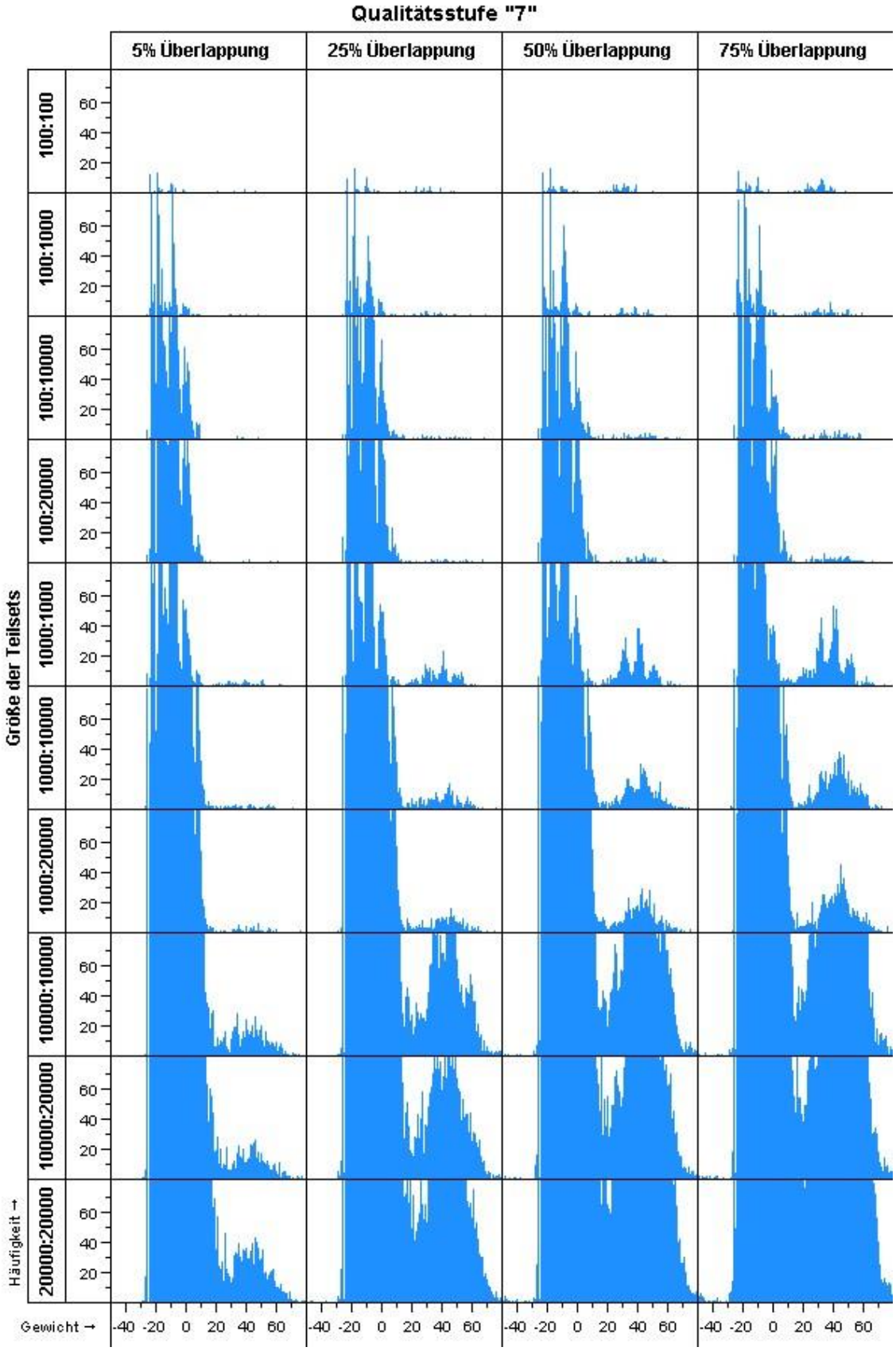

Qualitätsstufe "2"
5% Überlappung
25% Überlappung
50% Überlappung
75% Überlappung
Größe der Teilsets
100:100
100:1000
100:10000
100:20000
1000:1000
1000:10000
1000:20000
10000:10000
10000:20000
20000:20000
Häufigkeit →
Gewicht →
-40 -20 0 20 40 60
20 40 60

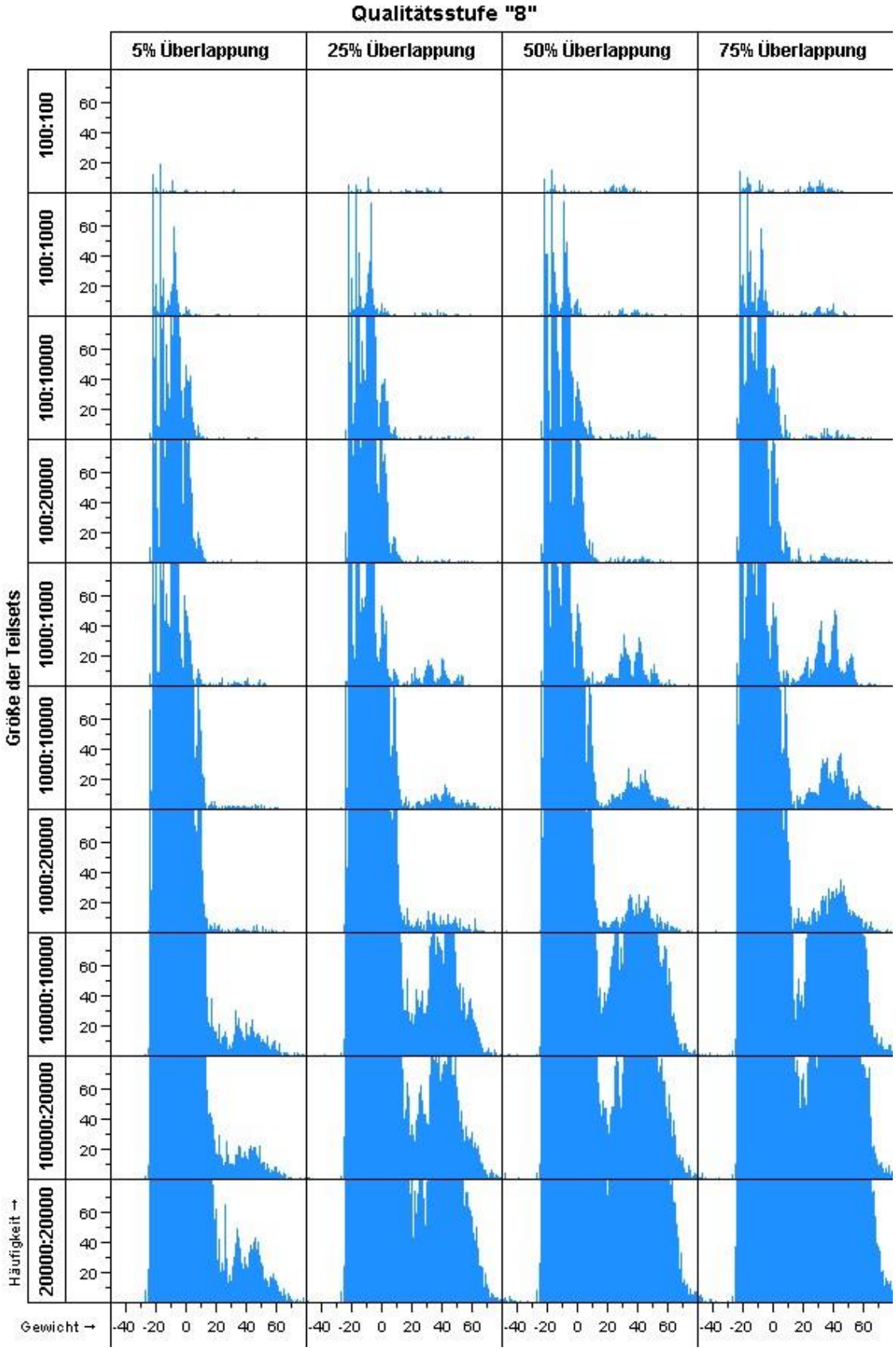

Qualitätsstufe "3"
5% Überlappung
25% Überlappung
50% Überlappung
75% Überlappung
Größe der Teilsets
100:100
100:1000
100:10000
100:20000
1000:1000
1000:10000
1000:20000
10000:10000
10000:20000
20000:20000
Häufigkeit →
Gewicht →
60
40
20
-40 -20 0 20 40 60

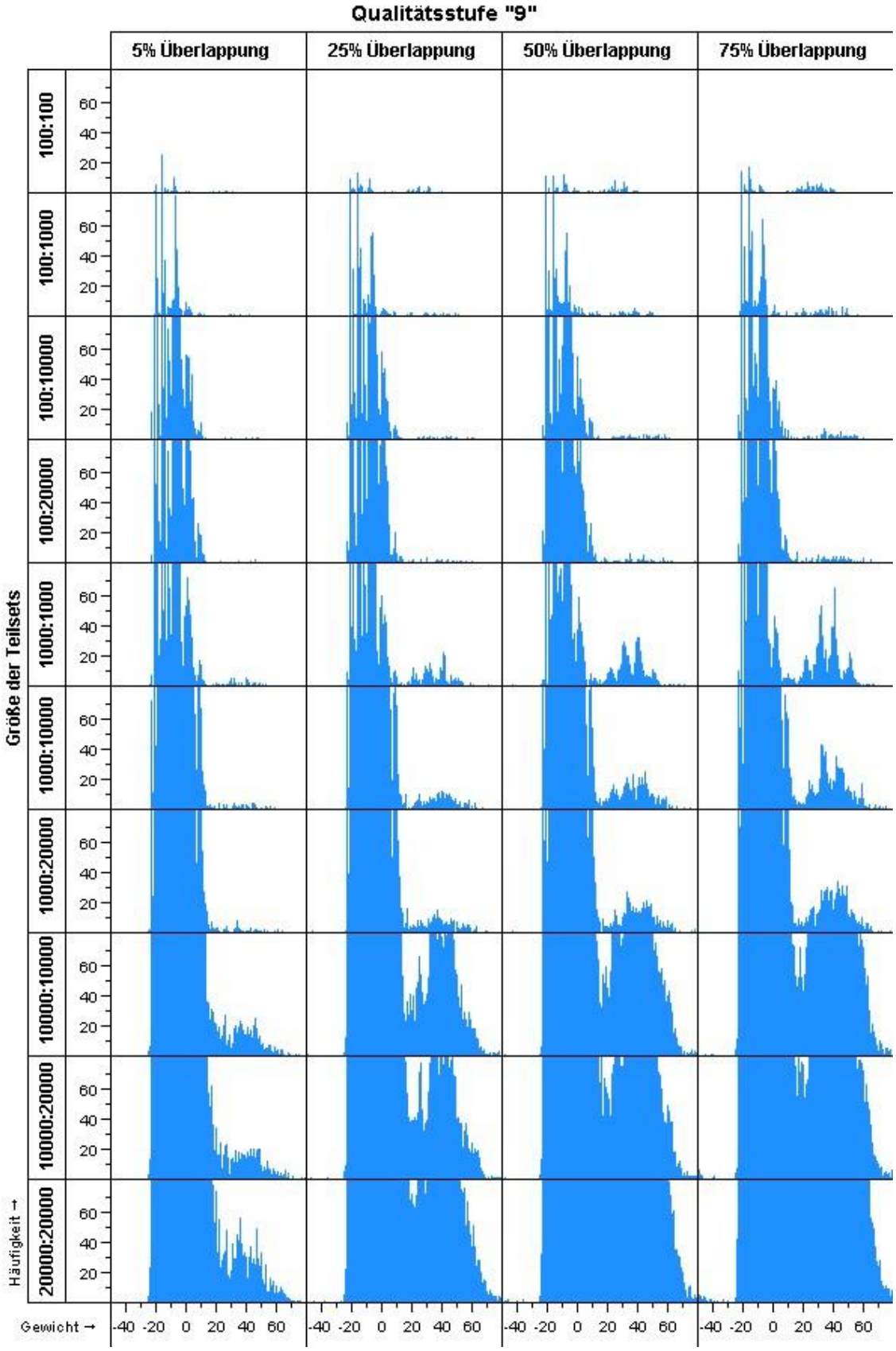

Qualitätsstufe "4"
5% Überlappung
25% Überlappung
50% Überlappung
75% Überlappung
Größe der Teilsets
100:100
100:1000
100:10000
100:20000
1000:1000
1000:10000
1000:20000
10000:10000
10000:20000
20000:20000
Häufigkeit →
Gewicht →
-40 -20 0 20 40 60
20 40 60

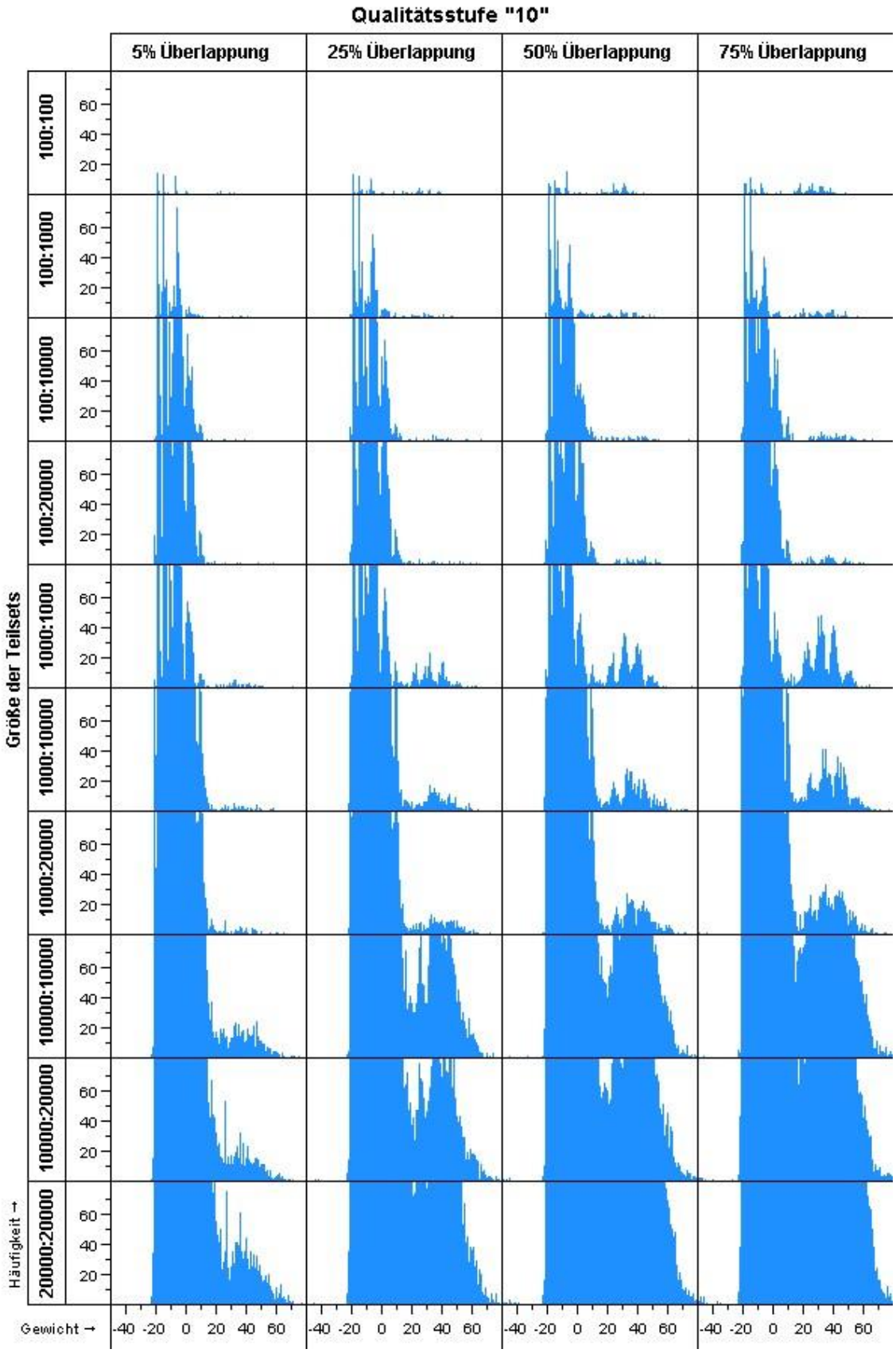

Qualitätsstufe "5"
5% Überlappung
25% Überlappung
50% Überlappung
75% Überlappung
Größe der Teilsets
100:100
100:1000
100:10000
100:20000
1000:1000
1000:10000
1000:20000
10000:10000
10000:20000
20000:20000
Häufigkeit →
Gewicht →
60
40
20
-40 -20 0 20 40 60


Qualitätsstufe "6"
5% Überlappung
25% Überlappung
50% Überlappung
75% Überlappung
Größe der Teilsets
100:100
100:1000
100:10000
100:20000
1000:1000
1000:10000
1000:20000
10000:10000
10000:20000
20000:20000
Häufigkeit →
Gewicht →
-40 -20 0 20 40 60
20 40 60


Qualitätsstufe "7"
5% Überlappung
25% Überlappung
50% Überlappung
75% Überlappung
Größe der Teilsets
100:100
100:1000
100:10000
100:20000
1000:1000
1000:10000
1000:20000
10000:10000
10000:20000
20000:20000
Häufigkeit →
Gewicht →
-40 -20 0 20 40 60
20 40 60


Qualitätsstufe "8"
5% Überlappung
25% Überlappung
50% Überlappung
75% Überlappung
Größe der Teilsets
100:100
100:1000
100:10000
100:20000
1000:1000
1000:10000
1000:20000
10000:10000
10000:20000
20000:20000
Häufigkeit →
Gewicht →
-40 -20 0 20 40 60
20 40 60


Qualitätsstufe "9"
5% Überlappung
25% Überlappung
50% Überlappung
75% Überlappung
Größe der Teilsets
100:100
100:1000
100:10000
100:20000
1000:1000
1000:10000
1000:20000
10000:10000
10000:20000
20000:20000
Häufigkeit →
Gewicht →
-40 -20 0 20 40 60
20 40 60


Qualitätsstufe "10"
5% Überlappung
25% Überlappung
50% Überlappung
75% Überlappung
Größe der Teilsets
100:100
100:1000
100:10000
100:20000
1000:1000
1000:10000
1000:20000
10000:10000
10000:20000
20000:20000
Häufigkeit →
Gewicht →
-40 -20 0 20 40 60
20 40 60

Danksagung

Manchmal scheinen Träume unerreichbar. So war es für mich gerade zu Beginn des Studiums der Bioinformatik schwierig, mit der neuen Selbstverantwortung und den hohen Anforderungen, die das Studium mit sich brachte umzugehen. Programmierung war mir fremd und von Induktionsbeweisen hatte ich noch nicht einmal ansatzweise gehört.

Nach dem Grundstudium jedoch saßen die wichtigsten Inhalte und der Stress und die Furcht vor dem Versagen wich der Begeisterung. Es war auch diese Zeit, zu der ich mich am IBE als studentische Hilfskaft bewarb. Ein mir bislang neues Feld. Die Medizininformatik. Sowohl die spannenden Tätigkeiten als auch die hervorragende Betreuung während dieser Zeit veranlassten mich, nach Abschluss des Bioinformatik-Studiums eine Stelle als wissenschaftlicher Mitarbeiter am IBE in der Arbeitsgruppe für Medizininformatik anzunehmen. Es gab also ein neues Ziel, die Promotion zum Dr. rer. Biol.hum., doch zu dieser Zeit schien der Traum noch in weiter Ferne. Wie das Leben so ist, spielt es einem manchmal übel mit. Kurz vor Antritt der neuen Stelle verstarb ein nahes Familienmitglied, mein Bruder, weswegen ich mir aufgrund der neuen Situation nicht mehr sicher war, ob ich der Sache mental gewachsen war. Dank des Zuspruchs meiner Familie und meiner Freunde wurde die Krise jedoch überwunden, die Wunden heilten und ich fühlte mich immer mehr in meiner neuen Rolle als Nachwuchswissenschaftler bekräftigt. Hiermit möchte ich mich ausdrücklich bei Euch bedanken.

Doch nicht nur meinem persönlichen Umfeld gehört der Dank. Auch die Atmosphäre in der akademischen Umgebung war stets angenehm und ich kann über die Kollegen sowohl in der eigenen Arbeitsgruppe als auch des kompletten Institutes nur Gutes berichten.

Besonders möchte ich mich aber bei Herrn Stausberg bedanken, der mich für das Fach der Medizininformatik begeistern konnte, dem ich im Grunde genommen die Stelle als wissenschaftlicher Mitarbeiter zu verdanken habe und der mich stets mit vollstem Einsatz, was die Thematik dieser Arbeit anging, begleitet hatte. Die Jahre der gemeinsamen Zusammenarbeit sowohl als studentische Hilfskraft als auch als wissenschaftlicher Mitarbeiter werde ich in absolut positiver Erinnerung behalten.

Mit dieser Arbeit habe ich mir einen Traum erfüllt und ich möchte nochmals allen danken, die mir hierzu verholfen haben.

Eidesstattliche Versicherung

Nasseh, Daniel

Name, Vorname

Ich erkläre hiermit an Eides statt,

dass ich die vorliegende Dissertation mit dem Thema

Einsatz und Optimierung einer überwachten Klassifizierungsmethode im Kontext eines Privacy-Preserving-Record-Linkage

selbständig verfasst, mich außer der angegebenen keiner weiteren Hilfsmittel bedient und alle Erkenntnisse, die aus dem Schrifttum ganz oder annähernd übernommen sind, als solche kenntlich gemacht und nach ihrer Herkunft unter Bezeichnung der Fundstelle einzeln nachgewiesen habe.

Ich erkläre des Weiteren, dass die hier vorgelegte Dissertation nicht in gleicher oder in ähnlicher Form bei einer anderen Stelle zur Erlangung eines akademischen Grades eingereicht wurde.

Ort, Datum

Unterschrift Doktorand