Anwendungen des Record-Linkage in der sozialwissenschaftlichen Forschung

Antrittsvorlesung

Rainer Schnell

Institut für Soziologie
Universität Duisburg-Essen

5.11.2008

Bedarf nach Record Linkage

- In den Sozialwissenschaften stammen die meisten Daten aus Surveys.
- Diese leiden unter den bekannten Problemen, z.B. Nonresponse und Erinnerungsfehler.
- Die benötigte Daten sind aber oft bereits vorhanden, aber in getrennten Datenbanken.
- Die Zusammenführung verschiedener Daten über identische Objekte aus verschiedenen Datenbanken nennt man „Record-Linkage“ (RL).

Historische Entwicklung des Record-Linkage

- Formulierung als Bayes-Klassifikator durch Newcombe (1959)
- Formalisierung durch Fellegi und Sunter (1969)
- Entwicklung des EM-Algorithmus für RL durch Jaro (1989)

Datenverknüpfungen mit PIDs und Identifikatoren

- Ideal sind aus technischer Sicht einheitliche Identifizierungsnummern (PIDs): Sie sind eindeutig und (fast) fehlerfrei.
- In Europa verwenden Belgien, Schweden, Norwegen, Dänemark und Finnland PIDs.
- In Deutschland wie in vielen anderen Ländern gibt es keine PIDs.
- Statt dessen werden Identifikatoren wie Namen, Geburtsdatum, Wohnadresse als Verknüpfungsschlüssel verwendet.
- Diese sind einzeln nicht eindeutig, sondern nur in Kombination sinnvoll.

Probleme mit Identifikatoren

- Identifikatoren in Datenbanken sind oft fehlerbehaftet.
- Bei Nachnamen finden sich z.B. regelmäßig in 20% der Fälle Schreibfehler.
- Daher führt ein exakter Abgleich zu Verknüpfungsfehlern.
- Diese Verknüpfungsfehler können zu systematischen Ausfällen führen.
- Exakter: Sie sind oft nicht „missing completely at random“ (MCAR).

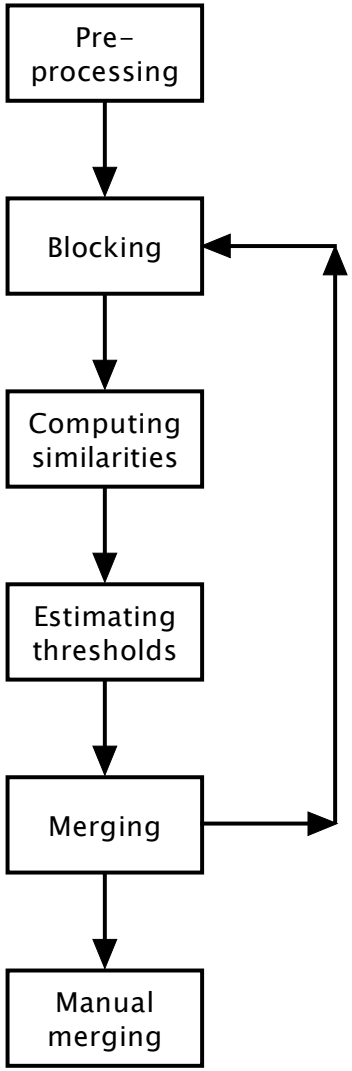
Techniken der Datenverknüpfung

- Deterministisches Record Linkage
  - Gefordert wird die exakte Übereinstimmung gewisser Verknüpfungsschlüssel.
  - Z.B. Nachname, Geburtstag, Geburtsmonat, PLZ.
  - Oft wird auch nur eine Mindestzahl an exakten Übereinstimmungen gefordert, z.B. 5 von 7.
- Distanzbasiertes Record Linkage
  - Hierbei wird die Forderung der exakten Übereinstimmungen aufgegeben.
  - Es werden Stringähnlichkeitsfunktionen verwendet, wie z.B. die Edit-Distanz.
- Probabilistisches Record Linkage
  - Die Programme basieren zumeist auf dem Fellegi/Sunter-Modell (1969).
  - Wegen fehlerbehafteter Strings werden auch hier Stringähnlichkeitsfunktionen verwendet.

Praktische Probleme und resultierende Projekte

- Bei der Arbeit mit Marburger Neonataldaten stießen wir auf ein Record-Linkage-Problem.
  - Heller u.a. (1999): Subpartuale Aziditätssteigerung und spätere neurologische und kognitive Entwicklung; in: Zeitschrift für Geburtshilfe und Neonatalogie, 2(203), S. 94
- Es gab kaum verfügbare Programme.
- Es gab kaum empirische Untersuchungen zu den technischen Eigenschaften.
- Dazu kamen Datenschutzprobleme.
- Daraus resultierten seit dem 1.1.2002 meine DFG-Projekte Filemerge, MTB und SAFELINK
- Die Hauptarbeiter:
  - Tobias Bachteler (Universität Duisburg-Essen)
  - Jörg Reiher (Fernuniversität Hagen)

Ablauf des Record-Linkage



Die Merge-Toolbox (MTB)

- MTB besteht aus drei Hauptmodulen:
  - dem „Pre-processing-Tool“
  - dem „Linkage-Modul“
  - dem „Manual Merge Modul“
- Als Ein- und Ausgabefiles dienen Statafiles oder Textfiles.
- Durch die Programmierung in Java läuft das Programm auf nahezu jedem Rechner.

Pre-Processing-Modul

Das Modul erlaubt unter anderem . . .

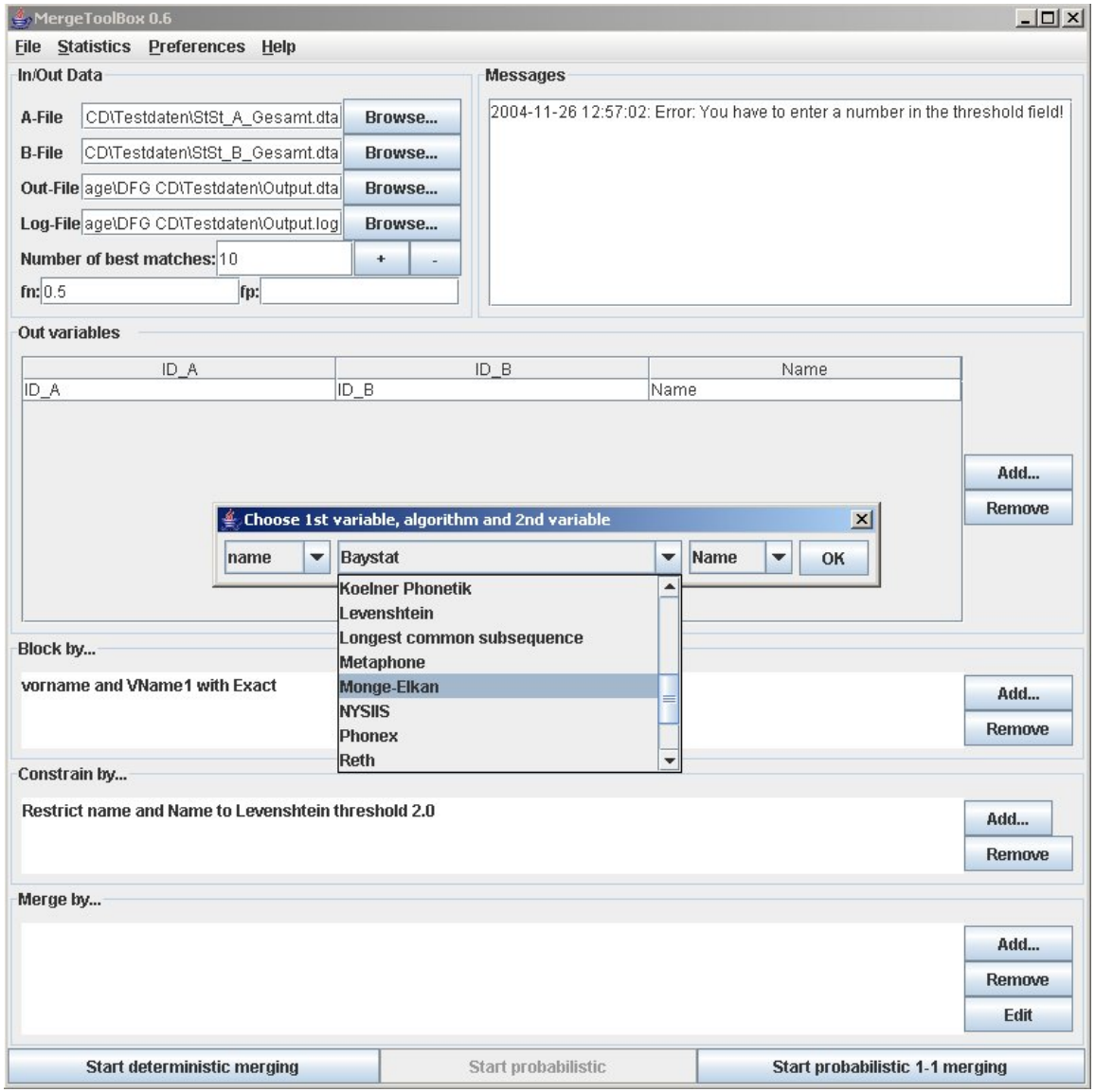
- eine doppelte Darstellung in getrennten Fenstern.
- die Darstellung einer Stichprobe der Records.
- die Standardisierung von Umlauten.
- das Entfernen von beliebigen Zeichenketten mittels „slack-Listen“.
- das „parsing“ von Doppelnamen in zusätzliche Datenzeilen.

Linkage-Modul

Das Modul erlaubt unter anderem . . .

- den exakten Abgleich.
- verschiedene Stringähnlichkeitsfunktionen und Phonetiken:
  - Edit-Distanzen: LCS, Levenshtein, Damerau-Levenshtein.
  - Bigramme und Trigramme.
  - Jaro's String Comparator und Erweiterungen.
  - Phonetik des statistischen Landesamtes Bayern.
  - Kölner Phonetik.
  - Soundex.
  - von Reth-Schek Phonetik.
- verschiedene Blocking-Schemata.

Screenshot des Link-Moduls



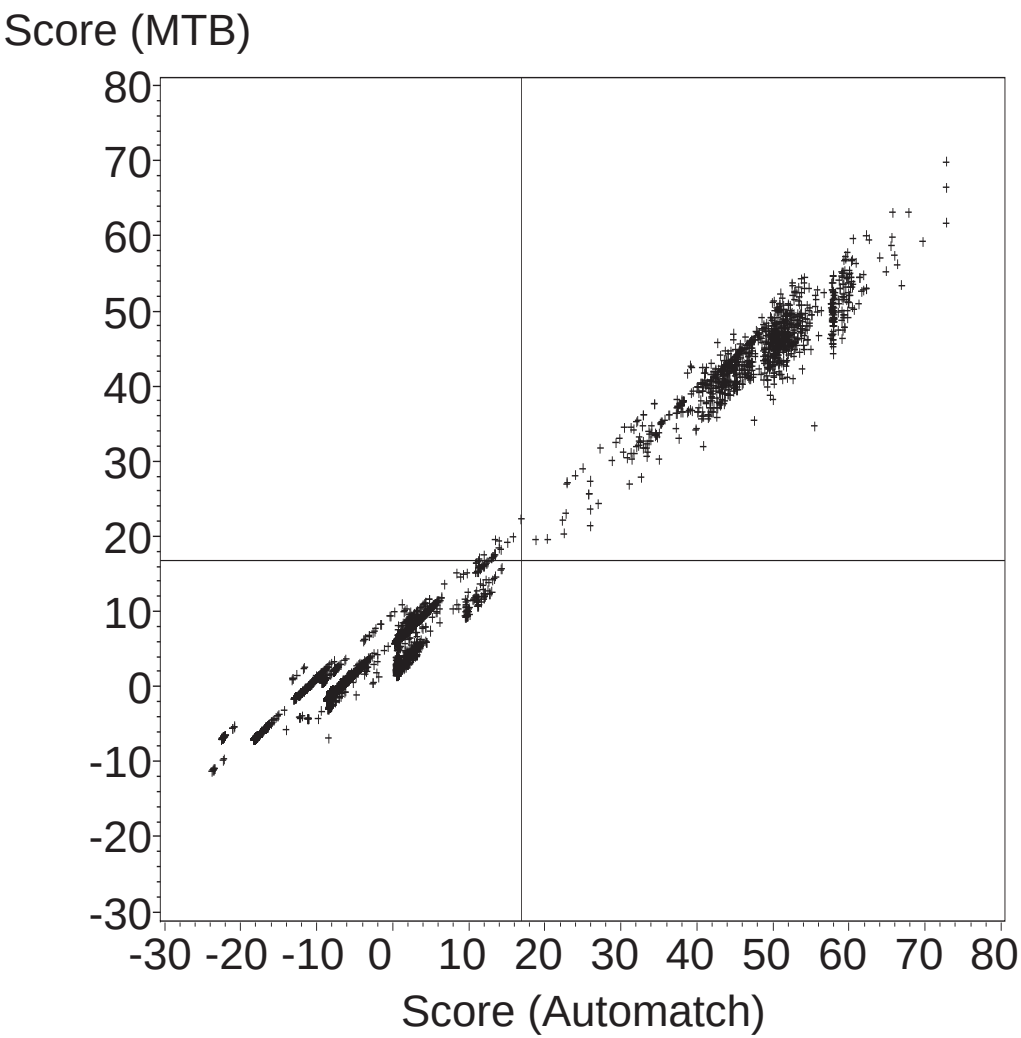
Manual-Merge-Modul

Zur Vereinfachung des manuellen Abgleichs erlaubt dieses Modul . . .

- unabhängig bewegbare und sortierbare Fenster.
- markieren und speichern von Record-Paaren als *match* oder *possible match* per Mausklick.
- die Subgruppenbildung für die Suche anhand von regulären Ausdrücken.
- die Subgruppenbildung anhand von maximaler Ähnlichkeit.
- die Bildung der Schnittmenge aus bereits gebildeten Subgruppen.

Validierung des Programms

- Validierungen von RL-Programmen sind schwierig, da in der Regel keine „true links“ bekannt sind.
- Der Vergleich mit Standardprogrammen ist daher hilfreich.
- Das in der Medizin am häufigsten verwendete Programm ist Automatch.
- Automatch ist nicht mehr käuflich zu erwerben, der Nachfolger (von IBM) für wissenschaftliche Anwendungen unbezahlbar.
- Wir haben daher mit realistischen Daten des Krebsregisters in Mainz einen Vergleich von MTB und Automatch vorgenommen.


Score (MTB)
80
70
60
50
40
30
20
10
0
-10
-20
-30
-30 -20 -10 0 10 20 30 40 50 60 70 80
Score (Automatch)

Ergebnis des Vergleichs MTB mit Automatch

- Größe der Datensätze: 16.000 vs 450.000 Fälle
- Dauer: 70 Minuten
- Korrelation der Scores: 0.974
- Konkordante Fälle: 99.28-99.95%
- Der manuelle Vergleich der diskordanten Paare spricht für MTB.

Veröffentlichungen zu MTB

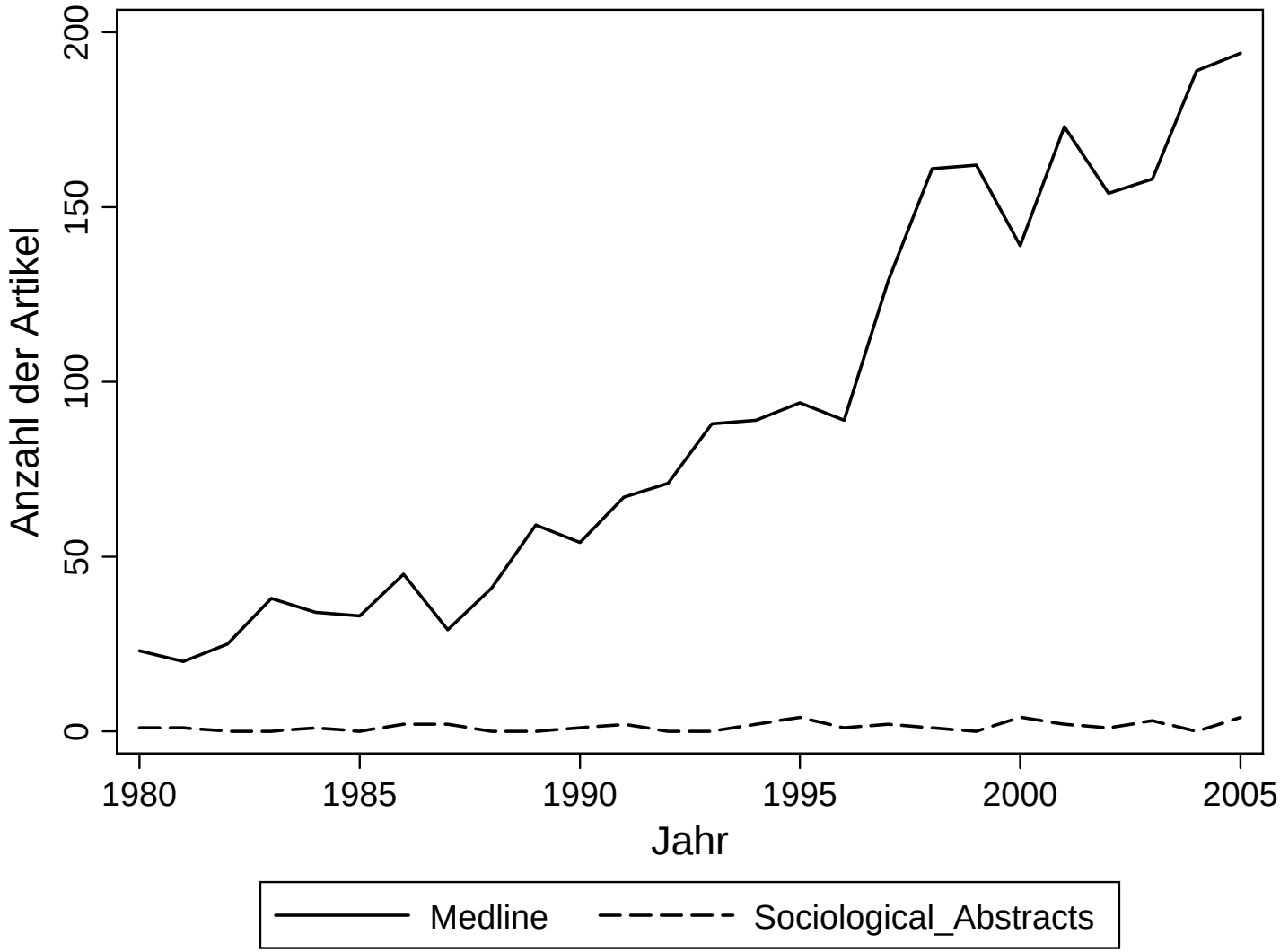
- Schnell,R./Bachteler,T./Reiher,J. (2004): A Toolbox for Record Linkage; in: Austrian Journal of Statistics, Vol. 33, No. 1-2, S.125-133.
- Schnell,R./Bachteler,T./Reiher,J. (2005): MTB: Ein Record-Linkage-Programm für die empirische Sozialforschung: ZA-Informationen, Heft 56, S.93-103.
- Hammer,G.P./Bachteler,T./Krtschil,A./ Reiher,J./Schnell,R. (2007): Die Verknüpfung epidemiologischer Datenbanken anhand personenidentifizierender Merkmale: Vergleich zweier stochastischer Record-Linkage Programme mit realistischen Daten, Poster auf dem Kongress der Deutschen Gesellschaft für Epidemiologie, Augsburg.

Bisherige Hauptanwendungsgebiete des Record-Linkage

Bislang wird Record-Linkage vor allem in folgenden Gebieten verwendet:

- Medizin, vor allem in der Epidemiologie und Versorgungsforschung
- Demographie
- Genealogie
- Data-Mining

Anzahl der Anwendungen pro Jahr



Anwendungen in Surveys I

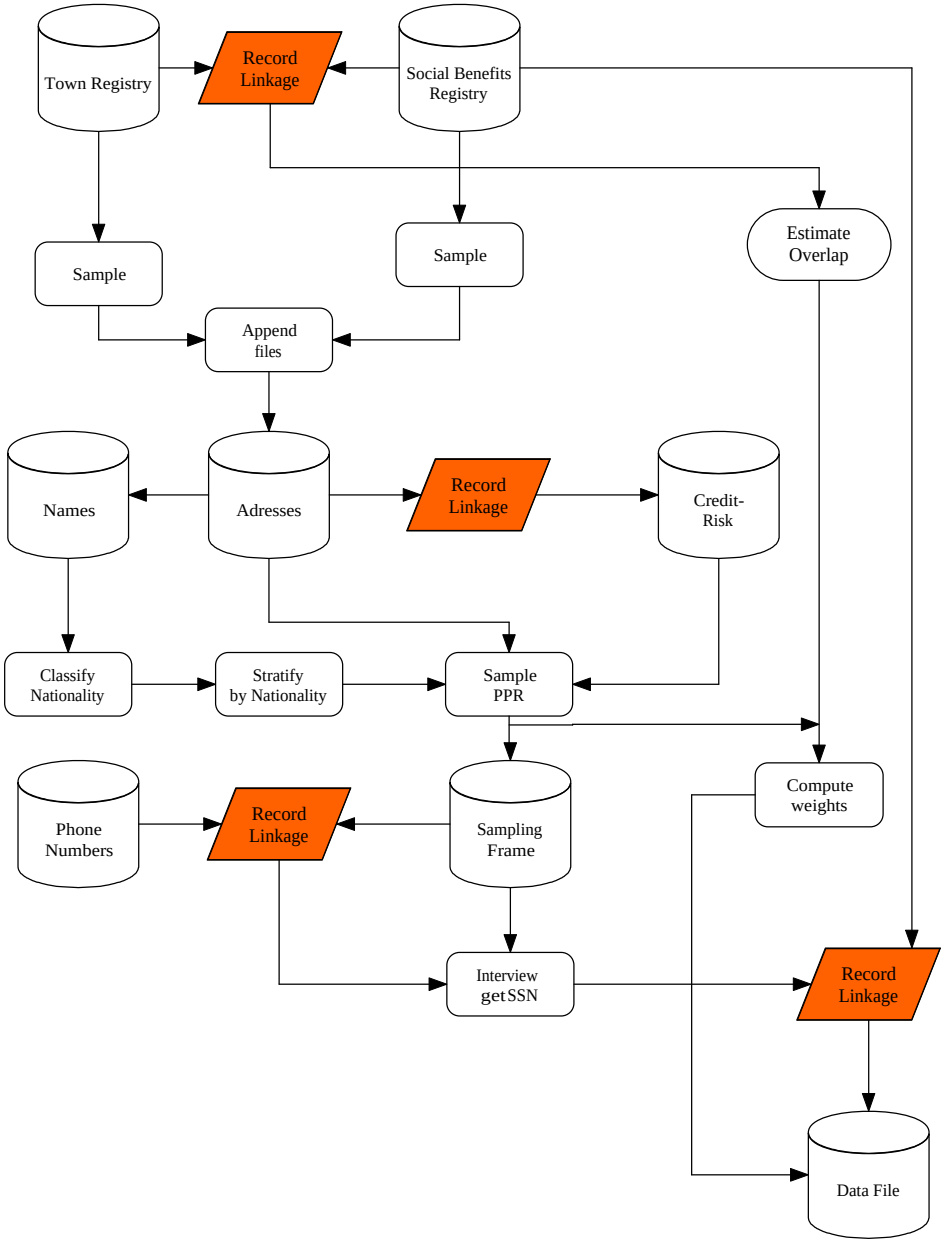
- Sampling-Frame-Konstruktion
  - Deduplikation bzw. Bestimmung des Overlaps in Surveys mit multiplen Frames
  - Entdeckung von Overcoverage und Undercoverage
- Nonresponse-Anwendungen
  - Bias-Entdeckung
  - Hilfe bei der Imputation durch Zuführung externer Variablen, z.B. Geodaten.
  - Hilfe bei der Imputation oder Gewichtung mit Registerdaten

Anwendungen in Surveys II

- Antwortvalidierung
  - Vergleich individueller Angaben mit Registerdaten, z.B. Impfstatus oder Einkommen
  - Haushaltsdefinition (Beispiel: Zensus)
- Panelkonstruktion
  - Retrospektive Panels (Beispiel: „Victorian Panel“ des ESRC)
  - Selbstgenerierte Codes

Beispiel Framekonstruktion: Stichprobenplan für Niedrigeinkommensbezieher

- 2006 sollte für das IAB-Panel „PASS“ ein Stichprobenplan entwickelt werden.
- Dabei sollten potentielle Leistungsbeziehern *vor* Eintritt des Ereignisses gezogen werden.
- Der Frame sollte mit wenigen Wochen Vorlauf erstellt werden.
- Der Entwurf (Schnell 2007) und der tatsächlich realisierte Plan verwendet mehrere, überlappende Sampling-Frames, die mit Record-Linkage verbunden werden.
- Schnell, R. (2007): Alternative Verfahren zur Stichprobengewinnung für ein Haushaltspanelsurvey mit Schwerpunkt im Niedrigeinkommens- und Transferleistungsbezug, in: Markus Promberger (Hrsg.): Neue Daten für die Sozialstaatsforschung. Zur Konzeption der IAB-Panelerhebung „Arbeitsmarkt und Soziale Sicherung“, IAB-Forschungsbericht Nr. 12/2007, S.33-59


Town Registry
Record Linkage
Social Benefits Registry
Sample
Sample
Estimate Overlap
Append files
Names
Adresses
Record Linkage
Credit-Risk
Classify Nationality
Stratify by Nationality
Sample PPR
Phone Numbers
Record Linkage
Sampling Frame
Compute weights
Interview getSSN
Record Linkage
Data File

Anwendungsbeispiel Panelkonstruktion: Führungskräftepanel

- Datenbasis: Leitende Männer (und Frauen) der Wirtschaft, 1950-2007 („Hoppenstedt“)
- Ablauf: Scannen, OCR, Datenextraktion, Record-Linkage
- Ergebnis: Unternehmensverflechtungen als dynamisches Netzwerk
- Matiaske,W./Nienhüser,W./Schnell,R. (2007): Paneluntersuchung zu institutionellen Netzwerken von Spitzenführungskräften der deutschen Wirtschaft, DFG-Antrag.

Anwendungsbeispiel Panelkonstruktion: Selbstgenerierte Codes

- In Panels werden bei sensitiven Themen selbstgenerierte Codes verwendet.
- Dies führt bei exaktem Match zu massiven Ausfällen durch Codefehler.
- Diese Ausfälle sind nicht MCAR.
- Daher werden fast immer manuelle fehlertolerante Abgleiche durchgeführt.
- Diese sind langwierig und fehlerbehaftet.
- Das Problem läßt sich durch Record-Linkage einfacher, schneller und fehlerfreier angehen.
- Schnell,R./Bachteler,T./Reiher,J. (2006): Anwendung statistischer Record-Linkage-Methoden auf selbst-generierte Codes bei Längsschnitterhebungen; in: ZA-Informationen, Heft 59, S. 128-142.

Reale sozialwissenschaftliche Anwendungen in den letzten zwei Jahren

- Validierung angegebener Telefonnummern mit Telefon-CDs (Defect-Projekt)
- Maschineller Lookup der Telefonnummern in Telefon-CDs bei EWA-Stichproben
- SGB-2-Evaluationen des IAB: Lookup der Sozialversicherungsnummern in den IAB-Dateien
- Link des IAB-Betriebspanels des IAB an Amadeus
- Link des IFO-Panels an Amadeus

„Privacy“

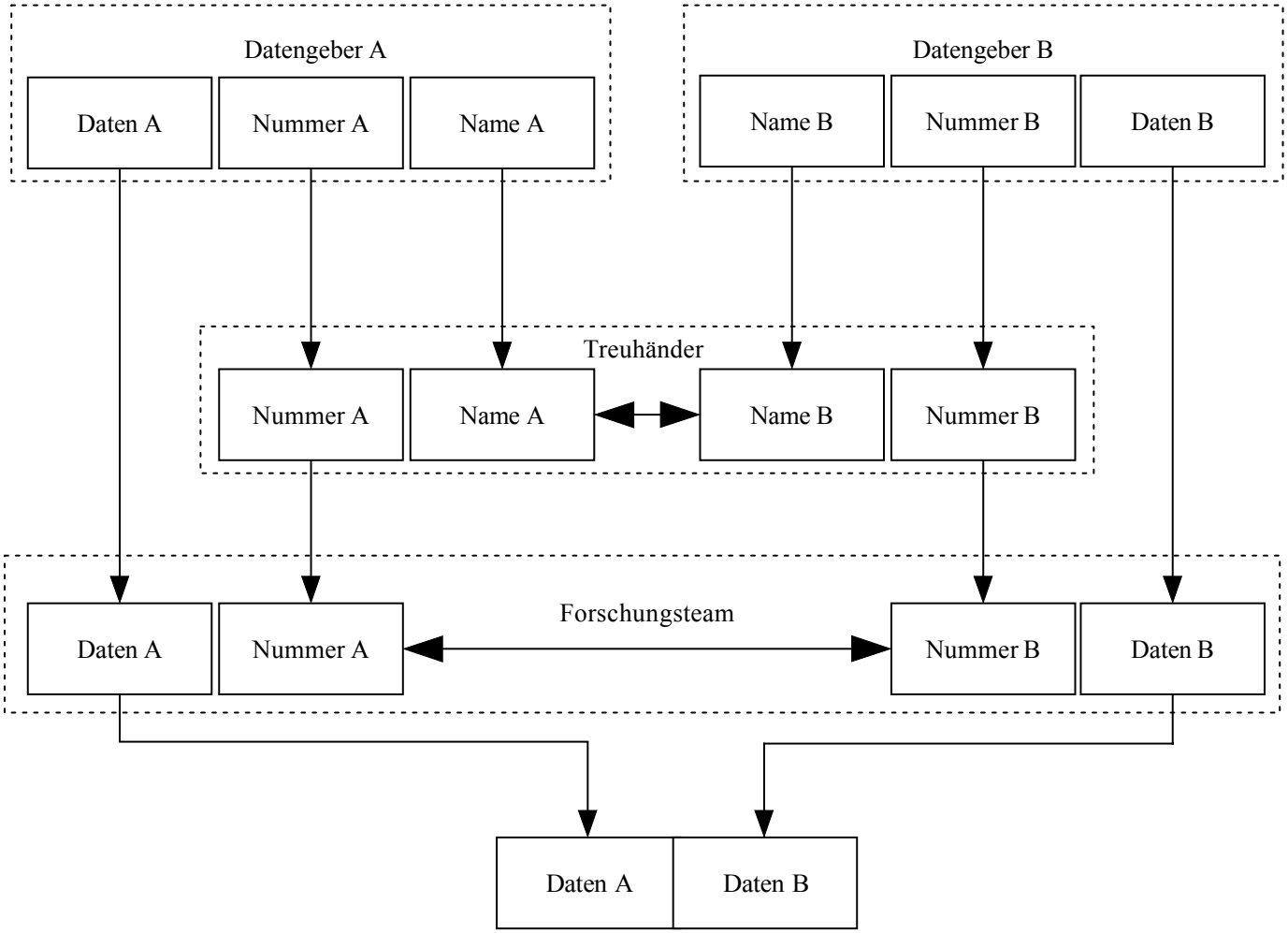
- Datenschutz wird vor allem in Europa stark betont.
- In der BRD sind die Dinge noch etwas interessanter:
  - Eine einheitliche Personenkennziffer wurde durch das Bundesverfassungsgericht verboten.
  - Die BRD besteht aus 16 Ländern, mit ingesamt 16 Datenschutzbehörden (+ Bundesdatenschutz) sowie 14 statistischen Landesämtern, alle mit eigenem Datenschutzbeauftragten und einem eigenen Gesetz.

Einwilligung der Betroffenen

Bei einem Survey braucht man eine schriftliche Einwilligung zum Linkage.

- Dies kann auch in der BRD durchgeführt werden.
- Zwischen 25 und 90% Einwilligung ist möglich.
- MCAR ist vermutlich nicht gegeben.

Lösung 1: Treuhänderlösungen



Probleme der Treuhandlösung

- Unüblich außerhalb der Medizin.
- In der Regel benötigt man einen Notar.
- Der Gebrauch von Klarschriftschlüsseln ist immer problematisch.
- Die meisten Anwendungen verwenden sehr schwache Verschlüsselungen.
- Es gibt aber starke Schlüssel (sogenannte HMACs).
- Fehler in einem einzigen Bit machen HMACs nutzlos für das Record-Linkage.

Lösung 2: Blindfolded trustee with encrypted keys

- Viel besser wäre eine Möglichkeit, die Daten „anonym“ zu verknüpfen.
- Dann würde die Weitergabe und Nutzung der identifizierenden Merkmale nicht mehr den Bestimmungen des Datenschutzrechtes unterliegen.
- Entscheidende Frage also: Wie können zwischen den Namen aus den beiden Datenbanken Stringähnlichkeitsfunktionen berechnet werden, ohne dass die Namen dabei offen gelegt werden?
- Tim Churches and Peter Christen (2004): Some methods for blindfolded record linkage; BMC Medical Informatics and Decision Making, 4, vol. 9; online published 28.6.2004

Protokoll: Hashsummen von Bigrammmengen I

1. A und B tauschen einen geheimen Zufallsschlüssel K.
2. A erzeugt für jeden Namen in N_A die Liste der Bigramme.
3. A erzeugt für jeden Namen in N_A die Potenzmenge dieser Liste.
4. Die Elemente jeder Teilmenge werden als eine Zeichenkette unter Hinzunahme von K mit einer Einweg-Hashfunktion transformiert und als $S_A = H_K(N_A)$ gespeichert.
5. A hängt an jede Zeile an:
   - Die ID des Originalnamens
   - Die Zahl der in der Teilmenge enthaltenen Bigramme
   - Die Zahl der Bigramme des Originalnamens
6. B führt die Schritte 2-5 für alle Namen in N_B ebenfalls aus.

Protokoll: Hashsummen von Bigrammmengen II

1. A und B schicken alles an C.
2. C gleicht für alle ID-Paare jeweils die Werte von S_A und S_B ab und berechnet mit Hilfe der Angaben über die Bigramm-Zahlen bei Übereinstimmung den Bigramm-Score.
3. C ermittelt für jedes ID-Paar den maximalen Bigramm-Score und löscht alles andere.

Beispiel: „Peter“

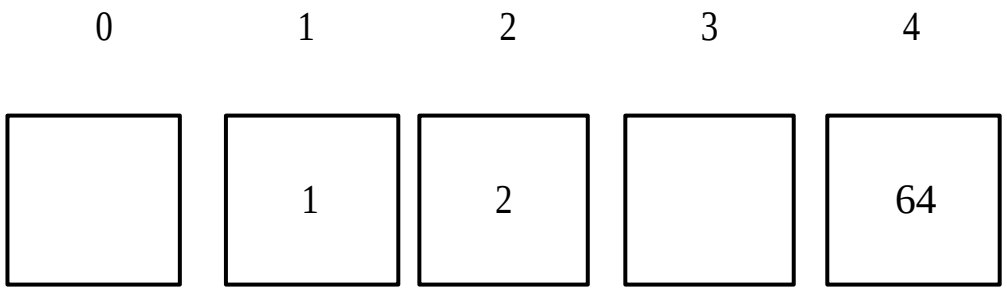
ID		S_A		
10	('er')	0a3be282870998f5	1	4
10	('et')	8898f53d6225f46b	1	4
10	('pe')	6fc83a87ee04335a	1	4
10	('te')	f2bcfb3d76d7fc03	1	4
10	('er', 'et')	f86abb0c84889d0e	2	4
10	('er', 'pe')	df99d8658d81651	2	4
10	('er', 'te')	edfb618d37ecfa1	2	4
10	('et', 'pe')	bd7ada000c2b900	2	4
10	('et', 'te')	fdcb71db96d2daa	2	4
10	('pe', 'te')	71322eeebabff9d	2	4
10	('er', 'et', 'pe')	8bf2788ef28443b7	3	4
10	('er', 'et', 'te')	c7e9a32e54ba33d5	3	4
10	('er', 'pe', 'te')	33287ce86aa02af0	3	4
10	('et', 'pe', 'te')	ecd7b151291f161e	3	4
10	('er', 'et', 'pe', 'te')	65e568493a08a34c	4	4

Lösung 3: SAFELINK

- Das Protokoll wurde bislang nicht veröffentlicht.
- Es ist kein Treuhänder notwendig.
- Es werden streng kryptographische Schlüssel verwendet.
- Das Protokoll ist fehlertolerant.
- Die Berechnungen erfolgen sehr schnell.
- Die Ergebnisse in Simulationen sind sehr ermutigend.

Was sind Hash-Funktionen?

- Eine Hash-Funktion ist eine Abbildung einer Menge von Werten in eine Menge von Speicheradressen.
- Die Werte können sowohl Zahlen als auch Strings sein.
- Die Speicheradressen sind Zahlen von 0 ... m-1.
- Beispiel:
  - Die Werte $(1, 2, 64)$ sollen gespeichert werden.
  - Die Hashfunktion sei $h(w) = w \mod 5$
  - Dann ist
    - h(1)=1
    - h(2)=2
    - h(64)=4



Was ist eine perfekte Hashfunktion?

- Eine perfekte Hash-Funktion besitzt keine Kollisionen, d.h. $h(w) \neq h(v)$ für alle $w \neq v$.
- Die Verteilung der Hashwerte soll über 0...m-1 gleichförmig sein.
- Hashfunktionen sollen sehr schnell berechenbar sein.

Was sind HMACs?

- HMACs sind „Message Authentication Codes“ basierend auf Einweg-Hashfunktionen.
- Bei einer Einweg-Hashfunktion kann aus dem Hash-Wert der Ursprungswert nicht mehr berechnet werden.
- Kryptographisch sinnvolle HMACs sind:
  - Chaotisch: Ähnlicher Input sollte zu völlig verschiedenen Hash-Werten führen. Im Idealfall verändert das Umkippen eines Bits in der Eingabe durchschnittlich die Hälfte aller Bits im resultierenden Hash-Wert.
  - Kollisionsresistent: Es soll unmöglich sein, zwei verschiedene Ausgangswerte mit dem gleichen Hashwert zu finden.

Ein Beispiel für HMACs mitSHA-1

The quick brown fox jumps over the lazy dog = 2fd4e1c6
7a2d28fc ed849ee1 bb76e739 1b93eb12

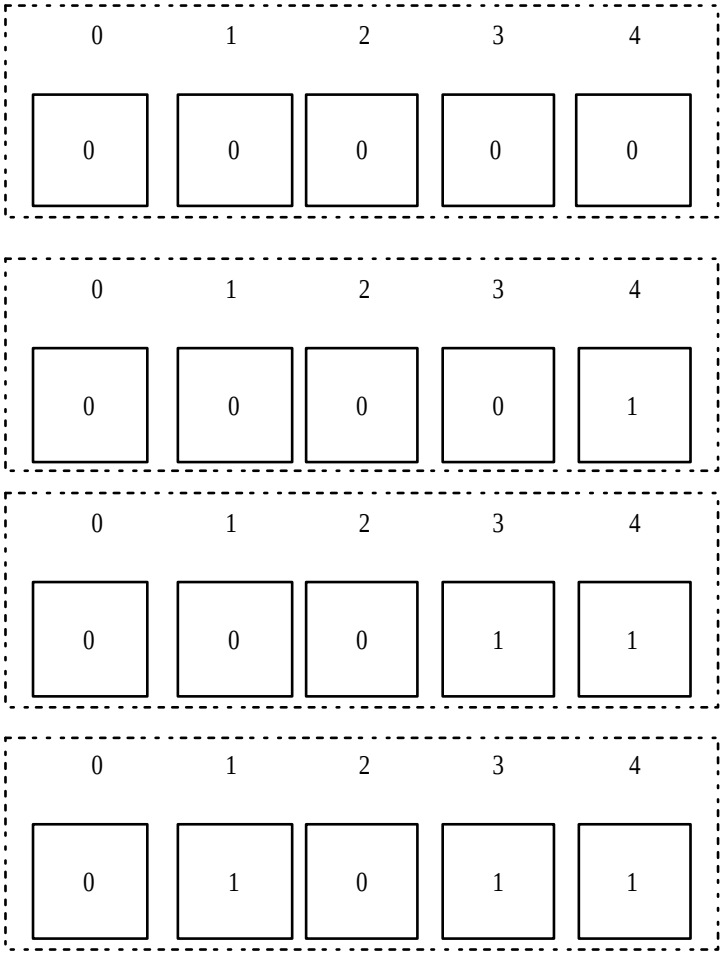
The quick brown fox jumps over the lazy cog = de9f2c7f
d25e1b3a fad3e85a 0bd17d9b 100db4b3

Was ist ein Bloomfilter?

- Ein Bloomfilter[1] ist eine Methode zur Hash-Kodierung der Elemente einer Menge $S = \{x_1, x_2, \ldots x_n\}$.
- Zunächst wird ein Bitarray der Länge m auf 0 gesetzt.
- Weiter werden k unabhängige Hash-Funktionen $h_1, h_2, \ldots, h_k$ definiert.
- Jede dieser Hash-Funktionen bildet einen Eingabewert in den Wertebereich $0, 1, \ldots, m-1$ ab.
- Für jedes $x \in S$ werden die Hash-Funktionen berechnet und die entsprechenden Bits auf 1 gesetzt.
- Treten die Hashwerte $h_i(x)$ mehrmals auf, bleibt das Bit $h_i(x)$ auf 1.

[1] Bloom,B.H. (1970): Space/Time Trade-offs in Hash Coding with Allowable Errors; in: Communications of the ACM Volume 13, Number 7, 422-426.

Beispiel für einen Bloomfilter, k=3, m=5, w=64



Step 1:
Set m bits to zero

Step 2:
Compute hash function 1:
h(x)= x mod 5 = h(64)= 64 mod 5 = 4
Set bit 4 = 1

Step 3:
Compute hash function 2:
h(x)= (x * 7) mod 5 = h(64)= (64 * 7) mod 5 = 448 mod 5=3
Set bit 3 = 1

Step 4:
Compute hash function 3:
h(x)= (x * 19) mod 5 = h(64)= (64 * 19) mod 5 = 1216 mod 5=1
Set bit 1 = 1

Eine Hash-Funktion für Bloom-Filter

- Die Wert für die i-te Hashfunktion
  $g_i(x) = (h_1(x) + i * h_2(x)) \mod p$
- wobei
  - p : Länge des Bitarrays m
  - h_1 : $SHA1$
  - h_2 : $MD5$
- SHA1: „Secure Hash Algorithm“ (NIST 1994), 160 bit key
- MD5: „Message Digest Algorithm 5“ (Rivest 1992); 128 bit key
- nach: Kirsch,A:/Mitzenbacher,M. (2006): Less Hashing, Same Performance: Building a Better Bloom Filter; in: Y. Azar und T. Erlebach (Hrsg.): ESA 2006, LNCS 4168, S.456-467

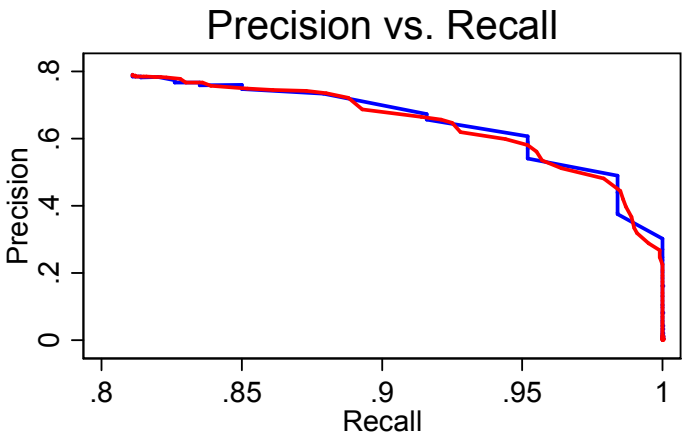
Ein neues Protokoll: SAFELINK

- Kein ähnliches Verfahren wurde bislang in der Literatur vorgeschlagen.
- SAFELINK ist innerhalb unseres Programms „Merge Toolbox“ implementiert: Es gibt ein funktionsfähiges Programm.
- SAFELINK entstand seit 2001 durch Diskussionen zwischen den Autoren im AFF-Projekt „Filemerge-1“ und den DFG-Projekten „Filemerge-2“ sowie „SAFELINK“ an der Universität Konstanz.
- Einen Hinweis auf Bloom-Filter zur Ähnlichkeitsberechnung im Allgemeinen verdanken wir Prof. Dr. Mathias Waldvogel, Konstanz.
- Für zahlreiche Diskussionen und Kritiken danken wir Prof. Dr. Ulrik Brandes, Konstanz.

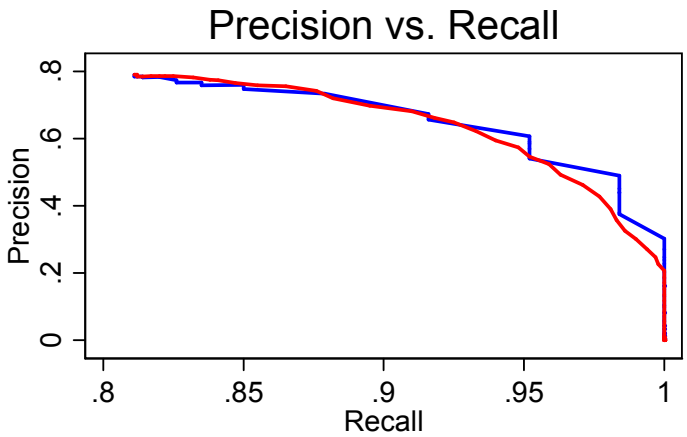
Das SAFELINK-Protokoll

1. A und B einigen sich auf die Länge m und die Zahl k.
2. Für jeden Namen i in $N.A$:
   - 2.1 A zerlegt den Namen i in Trigramme.
   - 2.2 A setzt für jedes Trigramm mit jeder der k Hashfunktionen im Bloomfilter$B.A.i$ das berechnete Bit.
3. A speichert alle Bloomfilter in einer Liste $B.A$
4. Entsprechend erstellt B für die Namen $N.B$ die Liste der Bloomfilter $B.B$.
5. A und B übermitteln die Listen $B.A$ und $B.B$ an eine dritte Partei C.
6. C vergleicht alle möglichen Paare an Bloomfiltern aus $B.A$ und $B.B$ bitweise und zählt dabei für jedes Paar die Zahl der gemeinsam auf 1 stehenden Bits.
7. Diese Zahl dividiert durch m ergibt eine Stringähnlichkeit für die den Bloomfiltern jeweils zugrunde liegenden Nachnamen.

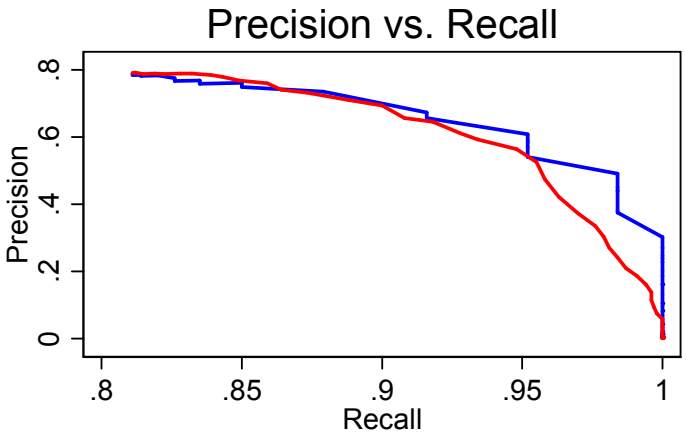
Performanz: SAFELINK



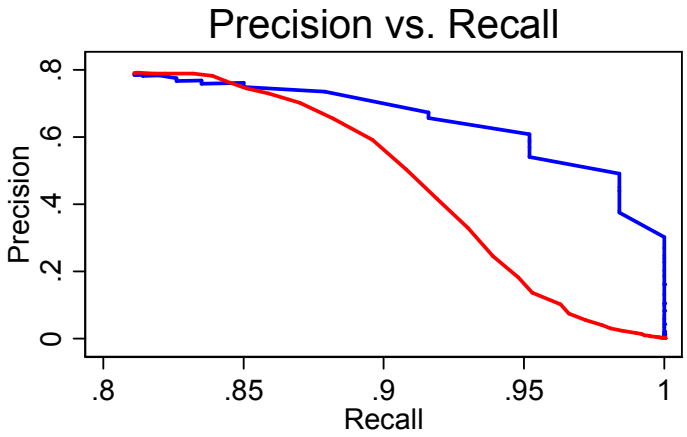
s001, Dice, 1000 bits, 5 Hashfunctions, Bloom: red



s001, Dice, 1000 bits, 25 Hashfunctions, Bloom: red



s001, Dice, 1000 bits, 50 Hashfunctions, Bloom: red



s001, Dice, 1000 bits, 100 Hashfunctions, Bloom: red

Warnungen

- Record-Linkage ist kein automatischer Prozess.
- Detailliertes Wissen über die datengenerierenden Prozesse ist notwendig.
- Detailliertes kulturelles Hintergrundwissen ist unverzichtbar.
- Die Datenaufbereitung erfordert fast immer die Beherrschung einer Skriptsprache wie AWK, Perl oder Python.
- Für jede Anwendungen sollten die Schwellenwerte neu geschätzt werden. Dies erfordert detaillierte Kenntnisse der Algorithmen und des Programms.

Hinweise

- Das Programm wird von uns frei zur Verfügung gestellt.
- In seiner vollen Leistungsfähigkeit ist es von Anfängern kaum einsetzbar.
- Support können wir nicht bieten – wir sind ein DFG-Forschungs-Projekt.
- Bei Projekten, die für eine Weiterentwicklung des Programms interessant sind, kooperieren wir bei gemeinsamen Veröffentlichungen (Krebsregister Bremen) oder bei einer Vollfinanzierung der Mitarbeiter (Krebsregister Hessen).

Zukunft des Projekts

- Veröffentlichung des neuen Algorithmus
- Empirischer Vergleich mit existierenden Algorithmen
- Zertifizierung des Programms
- Verbesserung der Pre-Processing-Routinen (Lexika, Hidden-Markov-Chains)
- Implementierung von Sparse-Matrix-Approximationen für das Blocken
- Bau eines massiv-parallelen Rechners für Blocking-Routinen
- Anbindung des Projekts an eine zentrale Infrastruktureinrichtung

Derzeitige Arbeitspapiere

- Schnell,R. (2008): Record-Linkage from a technical point of view; Expertise für das Gutachten der KVI-Kommission, Berlin, Oktober 2008 (abgeschlossen).
- Gramlich,T. (2008): Beschreibung der Verknüpfung der ifo-Konjunkturdaten mit der kommerziellen Firmendatenbank AMADEUS, Duisburg (abgeschlossen).
- Giersiepen,K. et al. (2008): Klartext oder Kontrollnummern? Ein Vergleich verschiedener Matchstrategien für den Datenabgleich zwischen epidemiologischen Krebsregistern und dem Mammografie-Screening, Bremen (in Überarbeitung).
- Schnell,R./Bachteler,T./Reiher,J. (2008): Privacy preserving approximate string comparisons by bloom filters for record linkage applications (in Bearbeitung).

Anwendungen der MTB-Programme

- Ifo: Konjunkturdaten
- IAB: Amadeus-Projekt
- IAB: PASS-Projekt
- Infas: Betriebsrätestichprobe Telefon-CDs
- Infas: Onomastische Stichprobe Hessen
- DFG: Deduplizierung Institute
- TU-Chemnitz: Framekonstruktion Piloten
- Krebsregister Hessen
- Krebsregister Bremen Mammographiedaten
- Uni Mainz: Evaluation des Krebsregisters NRW