

Verschiedene Verfahren zur Mikroaggregation als Klasse zur Anonymisierung von Mikrodaten

Betreuung: Prof. Dr. Thomas Augustin

Referent: Paul Fink

München, 8. Juni 2009

Gliederung

- 1 Motivation von Mikroaggregation
- 2 Durchführung von Mikroaggregation
 - Univariate Mikroaggregation
 - Fixed Size Mikroaggregation
 - Datengestützte Mikroaggregation
 - Multivariate Mikroaggregation
 - Fixed Size Mikroaggregation
 - Datengestützte Mikroaggregation
- 3 Kritik an k-Ward Algorithmus
 - Sicherheitsproblematik
 - Secure-k-Ward
 - Zeitaufwand bei Berechnung
 - Rekursiver k-Ward Algorithmus
- 4 Literatur

Zweck von Mikroaggregation

Mikrodaten enthalten oft sehr sensible Daten über Personen und/oder Firmen

⇒ Schutzmechanismen nötig zur Wahrung der Anonymität
Aber: Anonymisierung bedeutet auch Verlust an Information

Mikroaggregation als ein Weg zur Anonymisierung

Exkurs: Disclosure

Disclosure (engl.: Offenlegung): Schluss auf einzelne Werte bzw. kleine Wertintervalle nur über Ausnutzung der Daten

Generell gilt: Je geringer das Disclosure-Risk, desto sicherer das Verfahren

Aggregation aller Werte auf einen (z.b. Mittelwertbildung) bietet geringstes Disclosure-Risk

Problem: Kaum noch Informationsgehalt

Idee von Mikroaggregation

Mittelweg finden zwischen möglichst kleinem Disclosure-Risk und großem Informationsgehalt

2-Stufiges Verfahren:

- Aufteilung der Daten in homogene Gruppen
- Ersetzen der zu anonymisierenden Werte durch jeweilige Gruppenmittelwerte

Problem: Was bedeutet *homogene* Gruppen?

Wie ermittelt man diese?

k -Partitionierungs-Problem

Teile Gruppen so auf, dass folgende Bedingungen erfüllt sind:

- Mindestens k Beobachtungen in jeder Gruppe, aber nicht mehr als $2k$
- Minimierung der Streuung innerhalb der Gruppen

Konzept

- Festlegung der Gruppengröße
 - Genau vom Umfang k (fixed-size)
 - Mindestens vom Umfang k

Je größer k , desto größer der Grad der Anonymisierung und des Informationsverlustes

Generell $3 \leq k \leq 5$ vorher festgelegt

- Homogene Gruppen: Aufteilung in Gruppen, die die Streuung innerhalb der Gruppen minimiert
Verwandt mit dem Cluster-Problem bei Stichproben, hier jedoch minimale bzw. feste Gruppengröße

Streuungsbestimmung innerhalb der Gruppen schwierig bei mehrdimensionalen Daten

Überblick

Univariat, weil Daten nach nur einer Variable sortiert werden

Bei der Gruppenbildung werden 2 verschiedene Typen unterschieden:

- Fixed-Size-Mikroaggregation
- Datengestützte Mikroaggregation
 - Genetic Algorithmus (hier nicht behandelt)
 - k-Ward Algorithmus

Zur Sortierung werden 2 Konzepte verwendet:

- Single-Axis Sorting
- Individual Sorting

Single-Axis Sorting

Sortierung der Daten nach einer Variablen – kann auch ein Score von mehreren Variablen sein (z.B. z-Score)

Bilden von Gruppen der Größe k von beiden Enden der Daten

Wenn Umfang der Daten nicht Vielfaches von k , dann enthält größte Gruppe den Median der Daten

Ersetzen der Einzelwerte der Variablen durch Gruppenmittelwerte

Gut anwendbar, wenn alle zu ersetzenden Variablen mit der Sortierungsvariablen hoch korreliert sind

Individual Sorting

Grundannahme: Variablen sind alle unabhängig voneinander

Algorithmus zur Mikroaggregation von m Variablen:

i sei Index über die m Variablen $\implies i = 1 \dots m$

Schritt 1: Sortierung der Daten nach der i . zu aggregierenden Variable

Schritt 2: Gruppenbildung nach i . Variable wie bei Single-Axis Sorting

Schritt 3: Ersetzung der Einzelwerte nur für i . Variable

Schritt 4: Wenn $i = m$ verlasse Algorithmus,
sonst $i := i + 1$ und beginne wieder bei Schritt 1

Ward-Algorithmus

Setzt eine Sortierung der Daten voraus (Single-Axis oder Individual)

Methode zur Clusterung von Daten (Umfang n), benannt nach Joe H. Ward, Jr.

Kumulierendes hierarchisches Verfahren, das eine Reihe von Partitionierungen der Daten ergibt:

Erste Partitionierung besteht aus n Einzelgruppen, letzte Partitionierung umfasst eine Gruppe mit Umfang n

Prinzip: Iteratives Zusammenfügen von Gruppen, die am Ähnlichsten sind

Ähnlichkeit wird bestimmt über den Wert einer Abstandsfunktion; je kleiner dieser Wert, desto ähnlicher die Gruppen

Abstandsfunktion

Beschreibt den Informationsverlust, der beim Zusammenfügen zweier Gruppen entsteht

Dieser wird gemessen über die Varianz innerhalb einer durch Zusammenfügen entstehenden Gruppe

Entspricht für 2 Gruppen G_i (Umfang n_i) und G_j (Umfang n_j) folgender Formel:¹

$$d(G_i, G_j) = \frac{n_i n_j}{n_i + n_j} (\bar{x}_i - \bar{x}_j)^2$$

mit \bar{x}_i und \bar{x}_j Gruppenmittelwerte von G_i bzw. G_j

¹Vgl. Domingo-Ferrer and Mateo-Sanz: Practical Data-Oriented Microaggregation for Statistical Disclosure Control: S. 201

k -Ward Algorithmus

Ward-Algorithmus muss modifiziert werden, da in Cluster mit minimaler und maximaler Größe unterteilt werden soll
3 stufiger Ansatz von Domingo-Ferrer und Mateo-Sanz (1998):

Schritt 1: Bilde Gruppe der größten Datenpunkte vom Umfang k ;
Bilde Gruppe der kleinsten Datenpunkte vom Umfang k ;
Der Rest sind einelementige Gruppen;

Schritt 2: Anwendung Ward-Algorithmus bis alle Elemente zu einer Gruppe mit mindestens k Elementen gehören;
Einschränkung: Es dürfen nur Gruppen zusammengefügt werden mit Umfang kleiner k

Schritt 3: Wenn Gruppen mit Umfang $> 2k$, dann gehe zu Schritt 1 mit diesen Gruppen jeweils als Datenbasis,
sonst verlasse Algorithmus

Multivariate Mikroaggregation

2 verschiedene Herangehensweisen zur Mikroaggregation

- Projektion der Daten auf eine (künstliche) Variable;
Anwendung univariater Verfahren mit jener als
Sortierungsvariable
- Beibehaltung der multivariaten Struktur der Daten

Im Folgenden eine Fixed-Size Methode und eine Verallgemeinerung
des k-Ward Algorithmus

Fixed Size Mikroaggregation

Von Domingo–Ferrer und Mateo–Sanz vorgeschlagener Algorithmus:

Schritt 1: Bestimmung der Datenvektoren mit größtem Abstand

Schritt 2: Gruppierung der jeweils $k - 1$ nächsten Datenvektoren um die vorher in 1 Schritt bestimmten

Schritt 3: Umfang der Restvektoren (n^*):

$n^* < k$: Einzeln Hinzufügen zu der Gruppe mit dem kleinsten Abstand; verlasse Algorithmus

$k \leq n^* < 2k$: Bilde weitere Gruppe mit allen Restvektoren; verlasse Algorithmus

$n^* \geq 2k$: Wiederhole Algorithmus ab Schritt 1;
Als Datenbasis die Restvektoren

k-Ward Algorithmus

Lässt sich sehr leicht auch auf multivariate Daten übertragen

Problem: Was heißt *größte* bzw. *kleinste* Datenpunkte in Schritt 1?

Lösung: Bildung einer Abstands-Matrix

Diese umfasst alle Abstände eines Datenpunkts

Diese Matrix ist symmetrisch und hat 0 auf der Diagonale

Auswahl größter und kleinster Datenpunkte über maximalen Abstand

Danach Vorgehen wie im univariaten Fall

Sicherheitsproblematik

Minimierung der Streuung innerhalb von Gruppen

⇒ hohes Disclosure-Risk möglich

Annahme: Jede Gruppe sei (fast) vollkommen homogen

⇒ Schluss auf Originaldaten möglich

Zwar obige Annahme im Allgemeinen nicht bekannt,
Veröffentlichung dennoch sehr riskant

Lösungsvorschlag: Secure-k-Ward von Li, Zhu, Wang und Jajodia
(2002)

Notwendige Definitionen

Kenngößen zur Bestimmung der Sicherheit in einer Gruppe

Definition 3.1 (Tolerance Level ε):

Für jede Gruppe in mikroaggregierten Daten wird ein Ersetzungswert als "gefährlich" bezeichnet, wenn ein Datenvektor in einer solchen Gruppe einen Abstand kleiner als ein Skalar ε zu dem zu ersetzenden Wert hat, andernfalls als "sicher".

Definition 3.2 (Security Ratio γ):

Der Security Ratio beschreibt den Prozentsatz der Datenvektoren in einer Gruppe, die einen größeren Abstand als ε zu ihrem zu ersetzenden Wert haben, wobei ε das Tolerance Level ist

Idee von Secure-k-Ward

Nach Ausführung des k-Ward Algorithmus einen Sicherheitscheck einbauen

Dieser prüft, ob das errechnete $\gamma_i \geq \gamma_0$ in jeder Gruppe i gilt

Wenn es in einer Gruppe nicht gilt, wird ein größter und kleinster Wert gesucht, der die Sicherheitsbedingung erfüllt und gleichzeitig den Informationsverlust der Gruppe minimiert (Intra-Group Optimierung)

Im letzten Schritt wird geschaut, welcher Wert zu einer gruppenübergreifenden geringeren Abweichung vom Ersetzwert führt (Inter-Group Optimierung)

Intra-Group Optimierung I

\bar{x}_i bezeichne den Ersetzwert in einer Gruppe i und ε ein vorgegebenes Tolerance Level, dann ist das zugehörige Security Ratio

$$\gamma(\bar{x}_i, \varepsilon) = \frac{|S_i - S_i \cap I(\bar{x}_i, \varepsilon)|}{|S_i|}$$

mit $S_i = \{x_{ij} : j = 1 \dots n_i\}$ und $I(\bar{x}_i, \varepsilon) = \{x : |x - \bar{x}_i| < \varepsilon\}$

Ist $\gamma(\bar{x}_i, \varepsilon) \leq \gamma_0$ dann suche ein \overleftarrow{x}_i und \overrightarrow{x}_i

Lösung der nichtlinearen Optimierungsprobleme $\overleftarrow{\mathcal{P}}_i$ und $\overrightarrow{\mathcal{P}}_i$

Intra-Group Optimierung II

Eigenschaften von $\overleftarrow{\mathcal{P}}_i$ zur Ermittlung des kleineren Werts:

$$\text{Minimierung von } L(\overleftarrow{x}_i) = \frac{\sum_{j=1}^{n_i} (x_{ij} - \overleftarrow{x}_i)^2}{\sum_{j=1}^{n_i} (x_{ij} - \bar{x}_i)^2}$$

unter den Nebenbedingungen

- $\gamma(\overleftarrow{x}_i, \varepsilon) \geq \gamma_0$
- $\overleftarrow{x}_i \leq \bar{x}_i$

Existenz einer optimalen Lösung gesichert

Analoges Vorgehen für $\overrightarrow{\mathcal{P}}_i$

Inter-Group Optimierung I

Nun muss noch ausgewählt werden, welche der \overleftarrow{x}_i bzw. \overrightarrow{x}_i als Ersetzungswerte genommen werden

Dies geschieht durch die Minimierung der Overall-Abweichung vom

Ersetzwert $s = \left| \sum_{i=1}^{g'} u_i \cdot \overleftarrow{\Delta}_i + v_i \cdot \overrightarrow{\Delta}_i \right|$

unter Nebenbedingungen

- $u_i, v_i \in \{0, 1\}$
- $u_i + v_i = 1$

mit $\overleftarrow{\Delta}_i = (\overleftarrow{x}_i - \bar{x}_i) \cdot n_i$ und $\overrightarrow{\Delta}_i = (\overrightarrow{x}_i - \bar{x}_i) \cdot n_i$

Optimierungsproblem ist NP-schwierig

Inter-Group Optimierung II

Heuristische Methode zur Lösung des Problems:

- 1 $s^+ \leftarrow 0, s^- \leftarrow 0$
- 2 **for** $i \leftarrow 0$ **to** g' **do**
- 3 **if** $s^+ \leq -s^-$ **then** $u_i \leftarrow 1, v_i \leftarrow 0, s^- \leftarrow s^- + \overleftarrow{\Delta}_i$
- 4 **else** $u_i \leftarrow 0, v_i \leftarrow 1, s^+ \leftarrow s^+ + \overrightarrow{\Delta}_i$

mit $s = |s^+ + s^-| = \left| \sum_{i=1}^{g'} v_i \cdot \overrightarrow{\Delta}_i + \sum_{i=1}^{g'} u_i \cdot \overleftarrow{\Delta}_i \right|$

Dieses Problem ist dagegen linear und einfacher zu lösen

Zusammenfassung Secure-k-Ward

Kompromiss zwischen Sicherheit vor Disclosure und Informationsverlust

Methode eignet sich insbesondere bei sehr großen Datensätzen, da Overall-Abweichung vom Ersetzwert dann vernachlässigbar klein wird²

²Li, Zhu, Wang und Jajodia (2002)

Komplexität des k-Ward Algorithmus

Zuerst Berechnung aller Abstände der Datenpunkte untereinander

Komplexität der Abstandsberechnung: $\frac{n^2-n}{2} \hat{=} O(n^2)$

Komplexität der Berechnung bei Clusterung abhängig von
"Nettigkeit" der Daten:

- Im günstigsten Fall: $O(n)$
- Im schlechtesten Fall: $O(n^2)$

Gesamtkomplexität besteht also immer aus einer quadratischen
Komponente und einer günstigenfalls linearen Komponente

Idee

Fayyoumi und Oommen schlagen Algorithmus vor, der in Originaldaten vorhandene Abstände berücksichtigt und dadurch Komplexität der Clusterung reduziert

Dazu wird fest vorgegeben, in wie viele Teile der Datensatz gesplittet werden soll

Wenn das Verhältnis der Summe der Varianzen der einzelnen Subsets zur Varianz des gesamten Datensatzes eine vorher festgelegte Schranke überschreitet, dann wird dieser Algorithmus auf jedes einzelne Subset angewendet, ansonsten wird der k-Ward Algorithmus angewendet

Umsetzung

rekursiver k-Ward Algorithmus³ (kWR)

Input: *InSet* (sortierte Daten), θ (Gewählte Schranke), *J* (Splitting-Parameter)

Output: *OutSet* (Mikroaggregierte Daten)

- 1: Teile *InSet* in *J* disjunkte *InSet*₁, ... *InSet*_{*J*} auf
- 2: **if** $\theta > \frac{\sum_{i=1}^J \text{Var}(\text{InSet}_i)}{\text{Var}(\text{InSet})}$ **then**
- 3: call k-Ward (*InSet*)
- 4: **return** *OutSet*
- 5: **else**
- 6: **for** $i = 1 \leftarrow J$ **do**
- 7: call kWR(*InSet*_{*i*})
- 8: **end for**
- 9: **end if**
- 10: **return** *OutSet* = *OutSet*₁ \cup ... \cup *OutSet*_{*J*}
- 11: **end** kWR

³Vgl. Fayyoubi and Oommen: S. 330

Zusammenfassung rekursiver k-Ward

Vorteile:

- Aufsplittung im Voraus halbiert mindestens Berechnungszeit
- Ausnutzung einer schon in den Daten vorhandenen Struktur

Nachteile:

- Willkürliche Aufsplittung der Daten möglich
- Bestimmung von *richtigem* θ schwierig

Vielen Dank für Ihre Aufmerksamkeit

Literaturverzeichnis I

- Ward, J.H.: Hierarchical grouping to optimize an objective function. *Journal of the American Statistical Association*, **58** (1963) 236-244
- Mateo-Sanz, J.M., Domingo-Ferrer, J.: A comparative study of microaggregation methods. *Questiio* **22** (1998) 511-526
- Domingo-Ferrer, J., Mateo-Sanz, J.M.: Practical data-oriented microaggregation for statistical disclosure control. *IEEE Transactions on Knowledge and Data Engineering* **14** (2002) 189-201
- Domingo-Ferrer, J., Oganian, A., Torres, A., Mateo-Sanz, J.M.: On the Security of Microaggregation with Individual Ranking: Analytical Attacks. *International Journal of Uncertainty, Fuzziness and Knowledge/Based Systems*, **10** (2002) 477-491
- Fayyumi, E., Oommen, B.J.: On Optimizing the k-Ward Micro-aggregation Technique for Secure Statistical Databases. *Lecture Notes in Computer Science*, **4058** (2006) 324-335

Literaturverzeichnis II

- Li, Y., Zhu, S., Wang, L., Jajodia, S.: A privacy-enhanced microaggregation method. *FoKIS 02: Proceedings of the Second International Symposium on Foundations of Info. and Know. Sys.*, London, UK, Springer-Verlag (2002) 148159
- Schmid, M.: Estimation of a linear regression with microaggregated data (2007)