

Datenbereinigung von Zeitreihen

Von der Anomalieerkennung zur Anomalienreparatur

Jose Rodriguez Parra Flores
Klaus-Johan Ziegert

17. September 2019



Gliederung

- 1 Einführung
- 2 Grundlagen
- 3 Iterative Minimum Repairing
- 4 Evaluierung
- 5 Schluss

Überblick

- 1 Einführung
 - Motivation
 - Zielsetzung

- 2 Grundlagen

- 3 Iterative Minimum Repairing

- 4 Evaluierung

- 5 Schluss

Motivation: Problem

Messgeräte liefern unzuverlässige Daten

- GPS Tracker sind nahe von Gebäuden unzuverlässig
- Sensoren sind empfindlich gegenüber äußere Einflüsse
 - Z.B. starker Fall der Temperaturen bei einem Windzug



Abbildung: GPS-Tracking auf dem Campus der Tsinghua Universität [1]

Motivation: Anwendungen der Anomalieerkennung

Umgang von unzuverlässigen Daten mit Anomalieerkennung

① Unzuverlässige Datenpunkte entfernen

- Ausreißer werden entfernt 😊
- Entfernen aufeinanderfolgende Fehler machen Ergebnis unbrauchbar **oder** werden als solche ggf. nicht entfernt ☹️

② Unzuverlässige Datenpunkte reparieren

- Einzelne Ausreißer werden leicht korrigiert 😐
- Aufeinanderfolgende Fehler werden zu stark verändert (In der Praxis liegen die Messungen nahe bei den korrekten Werten) 😐

Motivation: Problemerweiterung

Hinzunahme von korrekt markierten Werten

- ① Markierung durch den Benutzer
 - Z.B. markiert der Benutzer in beliebigen Zeitabständen seinen aktuellen Standort
- ② Präzise Messgeräte liefern in längeren Zeitabstände korrekte Werte



Überblick

- 1 Einführung
 - Motivation
 - Zielsetzung

- 2 Grundlagen

- 3 Iterative Minimum Repairing

- 4 Evaluierung

- 5 Schluss

Zielsetzung

Ziel der Arbeit

- ① Berücksichtigung der markierten Werte in der Anomalieerkennung
 - Aufeinanderfolgende Fehler sollen besser abgeschätzt werden
- ② Anomalienreparatur mit den Minimum-Change-Prinzip vereinbaren
 - Keine drastische Veränderungen der Messwerte
- ③ Neue Anomalienreparatur hinsichtlich Berechnungslaufzeit, Ergebnisgenauigkeit usw. optimieren
- ④ Neue Anomalienreparatur mit unterschiedlichen Einstellungen mit weitere Verfahren empirisch vergleichen

Überblick

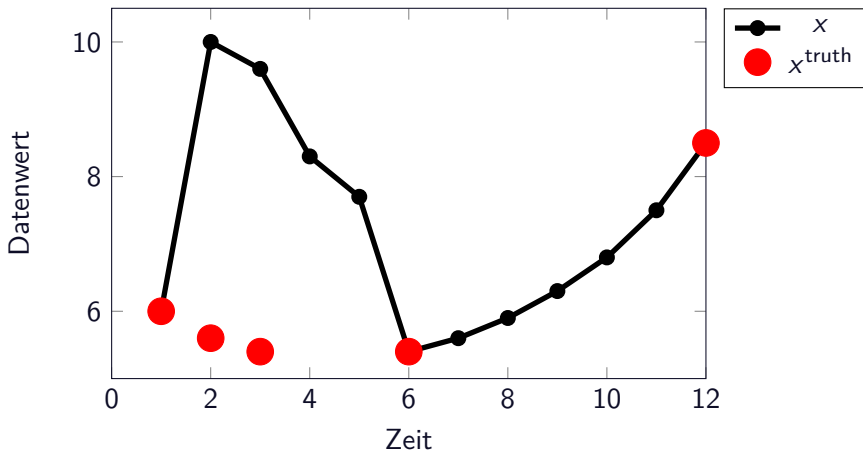
- 1 Einführung
- 2 Grundlagen
 - Problemstellung
 - Reparatur durch Anomalieerkennung
 - Andere Reparatur Methoden
- 3 Iterative Minimum Repairing
- 4 Evaluierung
- 5 Schluss

Problemstellung

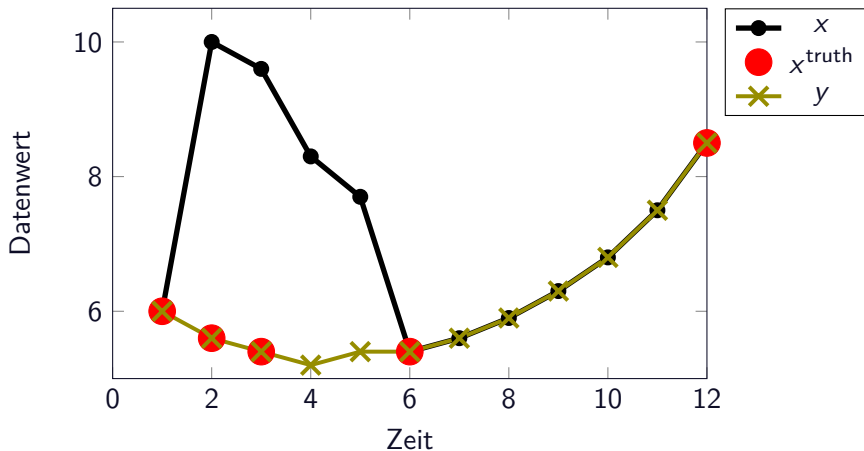
Zeitreihenreparatur

- Gegeben:
 - Unzuverlässige Messung $x = x[1], \dots, x[n]$
 - Unvollständige, aber dafür ausschließlich korrekte Messung x^{truth}
- Nur in der Evaluierung: vollständige, korrekte Messung $x^{\text{truth}*}$
- Gesucht:
 - Reparatur y mit minimalen RMS-Fehler $\Delta(x^{\text{truth}*}, y)$
 - $\Delta(x^{\text{truth}*}, y) = \sqrt{\frac{1}{n} \sum_{i=1}^n (x_i^{\text{truth}*} - y_i)^2}$

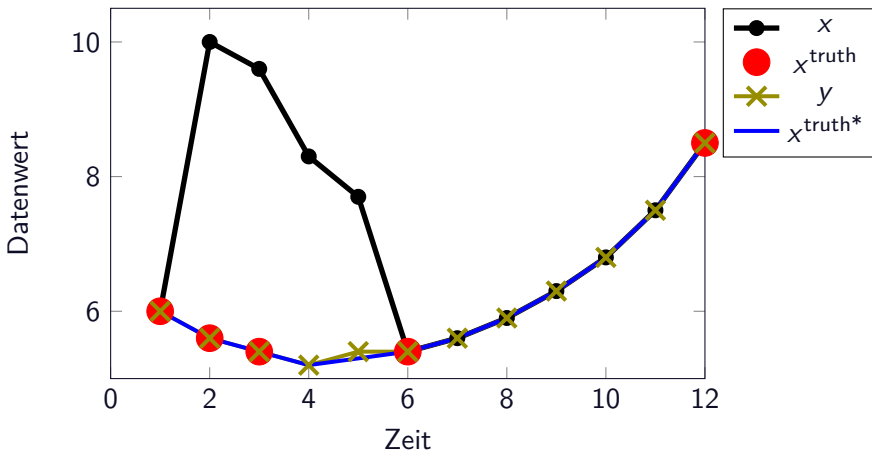
Problemstellung: Beispiel Eingabe



Problemstellung: Beispiel Eingabe & Reperatur



Problemstellung: Beispiel Eingabe & Reparatur



Problemstellung: Beispiel Zahlen & Bewertung

Zeitreihen vom Beispiel

- $x = \{6, 10, 9.6, 8.3, 7.7, 5.4, 5.6, 5.9, 6.3, 6.8, 7.5, 8.5\}$
- $x^{\text{truth}} = \{6, 5.6, 5.4, _, _, 5.4, _, _, _, _, _, 8.5\}$
- $y = \{6, 5.6, 5.4, \underline{5.2}, \underline{5.4}, \underline{5.6}, \underline{5.9}, \underline{6.3}, \underline{6.8}, \underline{7.5}, 8.5\}$
- $x^{\text{truth}*} = \{6, 5.6, 5.4, \underline{5.2}, \underline{5.3}, 5.4, \underline{5.6}, \underline{5.9}, \underline{6.3}, \underline{6.8}, \underline{7.5}, 8.5\}$

Bewertung des Beispiels

$$\Delta(x^{\text{truth}*}, y) =$$

$$\sqrt{\frac{1}{12} ((6 - 6)^2 + \dots + (5.3 - 5.4)^2 + \dots + (8.5 - 8.5)^2)} \approx 0.03$$

Überblick

- 1 Einführung
- 2 Grundlagen
 - Problemstellung
 - Reparatur durch Anomalieerkennung
 - Andere Reparatur Methoden
- 3 Iterative Minimum Repairing
- 4 Evaluierung
- 5 Schluss

Reparatur durch Anomalieerkennung

Anomalien

- Wikipedia: Abweichung von der Regel
- Werte x_i mit Abweichung τ (Bsp. $\tau = 2\sigma$):

$$|x_i - x_i^{\text{truth}}| > \tau$$

Reparatur durch Anomalieerkennung

Autoregressive Modell $AR(p)$

- Prädiktion aus den vorangegangenen p Werten:

$$x'_t = \sum_{i=1}^p \phi_i x_{t-i} + \epsilon_t$$

- Reparatur:

$$y_t = \begin{cases} x'_t & \text{falls kein Label und } |x'_t - x_t| > \tau \\ x_t & \text{sonst} \end{cases}$$

Reparatur durch Anomalieerkennung

Autoregressives exogenes Modell $ARX(p)$

- Exogene Variable y

$$y'_t = x_t + \sum_{i=1}^p \phi_i (y_{t-i} - x_{t-i}) + \epsilon_t$$

- y'_t Mögliche Reparatur:

$$y_t = \begin{cases} y'_t & \text{falls kein Label und } |y'_t - x_t| > \tau \\ y_t & \text{sonst} \end{cases}$$

ARX(1) Reparatur Beispiel

ARX(1) Reparatur Beispiel

$p = 1$, $\phi = 0.5$ und $\tau = 0.1$

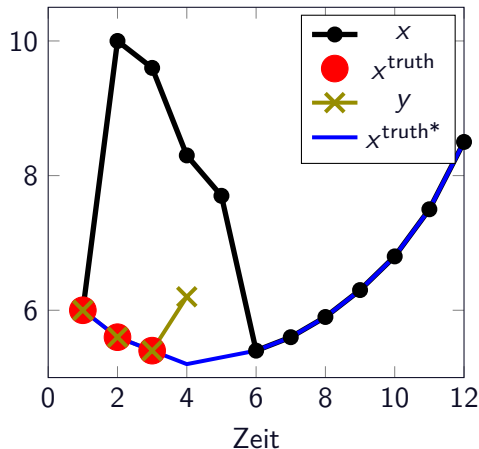
$$y'_4 = 8.3 + 0.5 \cdot (5.4 - 9.6)$$

$$y'_4 = 6.2$$

$$|6.2 - 8.3| = 2.1 > 0.1$$

$$y_4 = y'_4 = 6.2$$

Datenwert



Parameter Abschätzung

AR(1)

$$X_t = \phi X_{t-1} + \epsilon_t$$

Parameter Abschätzung

AR(1)

$$X_t = \phi X_{t-1} + \epsilon_t$$

Kleinste Quadrate Schätzung

$$\frac{\partial}{\partial \phi} \sum_{k=2}^n (X_k - \phi X_{k-1})^2 = 2 \sum_{k=2}^n (X_k - \phi X_{k-1})(-X_{k-1}) \quad (1)$$

Parameter Abschätzung

Kleinste Quadrate Schätzung

$$2 \sum_{k=2}^n (X_k - \phi X_{k-1})(-X_{k-1}) = 0$$

$$\sum_{k=2}^n (-X_k X_{k-1} + \phi X_{k-1}^2) = 0$$

$$-\sum_{k=2}^n X_k X_{k-1} + \sum_{k=2}^n \phi X_{k-1}^2 = 0$$

$$\sum_{k=2}^n \phi X_{k-1}^2 = \sum_{k=2}^n X_k X_{k-1}$$

$$\hat{\phi} = \frac{\sum_{k=2}^n X_k X_{k-1}}{\sum_{k=2}^n X_{k-1}^2}$$

Parameter Abschätzung $p = 1$

Beispiel

$$x = \{6, 10, 9.6, 8.3, 7.7, 5.4, 5.6, 5.9, 6.3, 6.8, 7.5, 8.5\}$$

$$y = \{6, 5.6, 5.4, 8.3, 7.7, 5.4, 5.6, 5.9, 6.3, 6.8, 7.5, 8.5\}$$

$$y - x = \{0, -4.4, -4.2, 0, 0, 0, 0, 0, 0, 0, 0, 0\}$$

$$\phi = \frac{(-4.2) \cdot (-4.4)}{(-4.2)^2 + (-4.4)^2} = 0.5$$

Überblick

- 1 Einführung
- 2 Grundlagen**
 - Problemstellung
 - Reparatur durch Anomalieerkennung
 - Andere Reparatur Methoden
- 3 Iterative Minimum Repairing
- 4 Evaluierung
- 5 Schluss

Glättungsverfahren

Gleitender Mittelwert

- Reparatur alle Werte durch y_j

$$y_j = \frac{1}{k} \sum_{i=1}^k x_{j-i} \quad j \in \{k, \dots, n\}$$

Glättungsverfahren

Gleitender Mittelwert

- Reparatur alle Werte durch y_j

$$y_j = \frac{1}{k} \sum_{i=1}^k x_{j-i} \quad j \in \{k, \dots, n\}$$

Exponentiell Gewichteten Gleitender Mittelwert (EWMA)

-

$$v_j = \sum_{i=0}^n (\beta - 1) \beta^i x_{j-i} \quad j \in \{k, \dots, n\}$$

- Effizient durch Dynamischesprogrammierung:

$$v_j = \beta v_{j-1} + (1 - \beta) x_j \quad \text{mit } V_0 = 0$$

Bedingung basiert Verfahren

SCREEN

- Die Reparatur erfolgt nur über zwei aufeinander folgende Werte
- Bedingung wie schnell ein Wert sich ändern kann
- gut für spikes aber nicht für Fehlern über längeren Zeitraum
-

Überblick

- 1 Einführung
- 2 Grundlagen
- 3 Iterative Minimum Repairing**
 - IMR
 - Optimierung 1: Matrix-Pruning IMR
 - Optimierung 2: Inkrementelle Berechnung
- 4 Evaluierung
- 5 Schluss

IMR Intuition

Intuitiver Ansatz von IMR

- ARX nutzt markierte Werte effizient, **aber** verändert die Werte zu drastisch.
- IMR Ansatz:
 - 1 Wende ARX an
 - 2 Wähle **einen** Reparaturwert mit minimalen Abstand zur Messung
 - 3 Wiederhole Prozedur bis aktuelle Reparatur sich nicht signifikant ändert
- Motivation: Reparierte Werte verbessern zukünftige Reparaturen

IMR = ARX + Minimum-Change-Prinzip

- 1: **Eingabe:** Messung x , markierte Werte x^{truth} , Ordnung p , Schwellenwert τ und max-num-iterations
- 2: **Ausgabe:** Reparatur y
- 3: $y^{(0)} \leftarrow \text{Initialize}(x, x^{\text{truth}})$
- 4: **for** $k \leftarrow 0$ **to** max-num-iterations **do**
- 5: $\phi^{(k)} \leftarrow \text{Estimate}(x, y^{(k)})$
- 6: $\hat{y} \leftarrow \text{Candidate}(x, y^{(k)}, \phi^{(k)})$
- 7: $y^{(k+1)} \leftarrow \text{Evaluate}(x, y^{(k)}, \hat{y})$
- 8: **if** $\text{Converge}(y^{(k)}, y^{(k+1)})$ **then**
- 9: **break**
- 10: **end if**
- 11: **end for**
- 12: **return** $y^{(k)}$

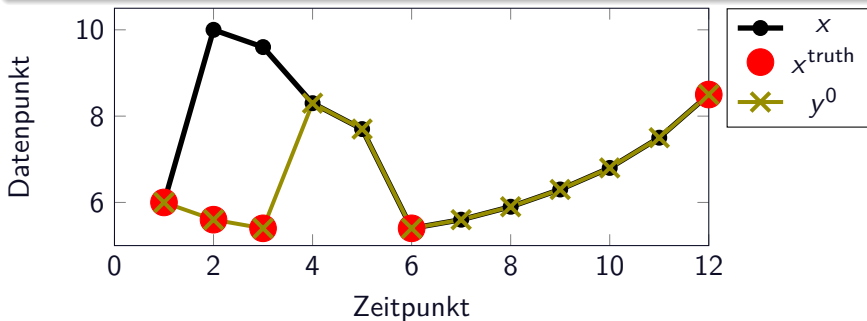
IMR: Initialisierung

- 1: **Eingabe:** Messung x , markierte Werte x^{truth} , Ordnung p , Schwellenwert τ und max-num-iterations
- 2: **Ausgabe:** Reparatur y
- 3: $y^{(0)} \leftarrow \text{Initialize}(x, x^{\text{truth}})$
- 4: **for** $k \leftarrow 0$ **to** max-num-iterations **do**
- 5: $\phi^{(k)} \leftarrow \text{Estimate}(x, y^{(k)})$
- 6: $\hat{y} \leftarrow \text{Candidate}(x, y^{(k)}, \phi^{(k)})$
- 7: $y^{(k+1)} \leftarrow \text{Evaluate}(x, y^{(k)}, \hat{y})$
- 8: **if** $\text{Converge}(y^{(k)}, y^{(k+1)})$ **then**
- 9: **break**
- 10: **end if**
- 11: **end for**
- 12: **return** $y^{(k)}$

IMR: Initialisierung

Initiale Reparatur

Initiale Reparatur $y^{(0)}$ ist Messung x und übernimmt die markierten Werte aus x^{truth}



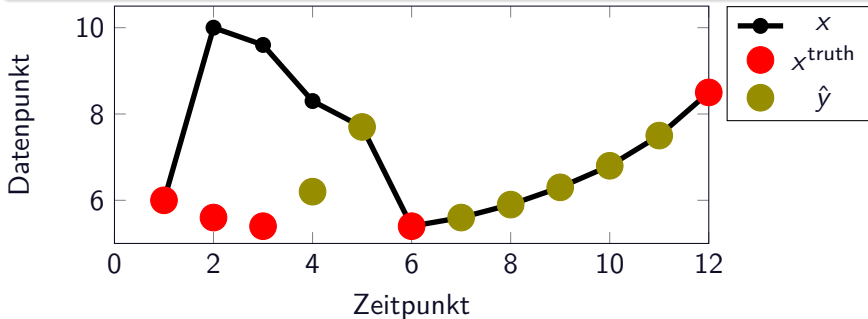
IMR: ARX auf aktuelle Reparatur anwenden

- 1: **Eingabe:** Messung x , markierte Werte x^{truth} , Ordnung p , Schwellenwert τ und max-num-iterations
- 2: **Ausgabe:** Reparatur y
- 3: $y^{(0)} \leftarrow \text{Initialize}(x, x^{\text{truth}})$
- 4: **for** $k \leftarrow 0$ **to** max-num-iterations **do**
- 5: $\phi^{(k)} \leftarrow \text{Estimate}(x, y^{(k)})$
- 6: $\hat{y} \leftarrow \text{Candidate}(x, y^{(k)}, \phi^{(k)})$
- 7: $y^{(k+1)} \leftarrow \text{Evaluate}(x, y^{(k)}, \hat{y})$
- 8: **if** $\text{Converge}(y^{(k)}, y^{(k+1)})$ **then**
- 9: **break**
- 10: **end if**
- 11: **end for**
- 12: **return** $y^{(k)}$

IMR: ARX auf aktuelle Reparatur anwenden

Kandidaten

- Parameterschätzung ϕ : aktuelle Reparatur $y^{(k)}$ wird als x^{truth} interpretiert.
- Kandidaten \hat{y} sind neue Reparaturwerte



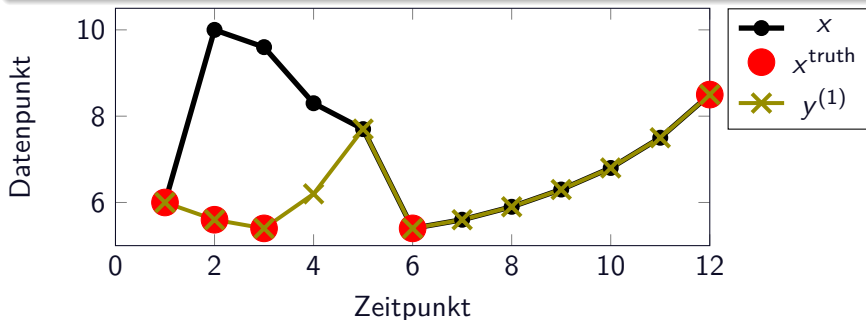
IMR: Minimum-Change

- 1: **Eingabe:** Messung x , markierte Werte x^{truth} , Ordnung p , Schwellenwert τ und max-num-iterations
- 2: **Ausgabe:** Reparatur y
- 3: $y^{(0)} \leftarrow \text{Initialize}(x, x^{\text{truth}})$
- 4: **for** $k \leftarrow 0$ **to** max-num-iterations **do**
- 5: $\phi^{(k)} \leftarrow \text{Estimate}(x, y^{(k)})$
- 6: $\hat{y} \leftarrow \text{Candidate}(x, y^{(k)}, \phi^{(k)})$
- 7: $y^{(k+1)} \leftarrow \text{Evaluate}(x, y^{(k)}, \hat{y})$
- 8: **if** $\text{Converge}(y^{(k)}, y^{(k+1)})$ **then**
- 9: **break**
- 10: **end if**
- 11: **end for**
- 12: **return** $y^{(k)}$

IMR: Minimum-Change

Minimum-Change

- Zu geringe Änderungen werden herausgefiltert $|y_i^{(k)} - \hat{y}_i| > \tau$
- Geringste Änderung zu Messung x wird als Kandidat ausgewählt



IMR: Terminierung

- 1: **Eingabe:** Messung x , markierte Werte x^{truth} , Ordnung p , Schwellenwert τ und max-num-iterations
- 2: **Ausgabe:** Reparatur y
- 3: $y^{(0)} \leftarrow \text{Initialize}(x, x^{\text{truth}})$
- 4: **for** $k \leftarrow 0$ **to** max-num-iterations **do**
- 5: $\phi^{(k)} \leftarrow \text{Estimate}(x, y^{(k)})$
- 6: $\hat{y} \leftarrow \text{Candidate}(x, y^{(k)}, \phi^{(k)})$
- 7: $y^{(k+1)} \leftarrow \text{Evaluate}(x, y^{(k)}, \hat{y})$
- 8: **if** $\text{Converge}(y^{(k)}, y^{(k+1)})$ **then**
- 9: **break**
- 10: **end if**
- 11: **end for**
- 12: **return** $y^{(k)}$

IMR: Terminierung

Terminierung

- Zwei Möglichkeiten der Terminierung:
 - Maximale Anzahl der Iterationen wird erreicht
 - Konvergenz: Neue Reparatur $y^{(k+1)}$ ist gleich aktuelle Reparatur $y^{(k)}$
- Allgemeine Konvergenzfrage ist noch offen

Überblick

- 1 Einführung
- 2 Grundlagen
- 3 Iterative Minimum Repairing**
 - IMR
 - Optimierung 1: Matrix-Pruning IMR
 - Optimierung 2: Inkrementelle Berechnung
- 4 Evaluierung
- 5 Schluss

Motivation von Matrix-Pruning IMR

Laufzeit- & Platzproblem

- Parameterschätzung beansprucht viel Zeit und Platz
- Matrizen V und Z bestehen aus $y_i^{(k)} - x_i$:
 - wenige markierte Werte vorhanden
 - markierte Werte häufig identisch zur Messung
 - Reparaturwerte ändern sich nicht signifikant
 - → dünnbesetzte Matrizen
- Matrix-Pruning: Löschen von Zeilen mit 0en

Matrix Pruning IMR Beispiel

Beispiel

Zeilen mit 0en in Z und entsprechende Zeile in V sind entfernbare:

$$Z = \begin{pmatrix} 0 \\ -4.4 \\ -4.2 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \quad V = \begin{pmatrix} -4.4 \\ -4.2 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \rightarrow Z_{mp} = \begin{pmatrix} -4.4 \\ -4.2 \end{pmatrix} \quad V_{mp} = \begin{pmatrix} -4.2 \\ 0 \end{pmatrix}$$

Überblick

- 1 Einführung
- 2 Grundlagen
- 3 Iterative Minimum Repairing**
 - IMR
 - Optimierung 1: Matrix-Pruning IMR
 - Optimierung 2: Inkrementelle Berechnung
- 4 Evaluierung
- 5 Schluss

Inkrementelle Berechnung (IMR-IC)

Intuition

- IMR Algorithmus berechnet ϕ^k in jede Iteration k

$$\phi^k \leftarrow \text{Estimate}(x, y^k)$$

- Minimum-Change-Prinzip, ein Wert r wird geändert

$$y_r^k \neq y_r^{k-1}$$

- Fast alle Werte in Z^k, Z^{k-1} und V^k, V^{k-1} bleiben unverändert

Inkrementelle Berechnung (IMR-IC)

Rekursive Formel

Sei $\phi^{(k)} = (A^{(k)})^{-1}B^{(k)}$ mit $A^{(k)} = (Z^{(k)})'Z^{(k)}$ und $B^{(k)} = (Z^{(k)})'V^{(k)}$

Fall: $1 \leq i \leq p$

$$a_{ii}^{(k)} = a_{ii}^{(k-1)} + \begin{cases} 0 & \text{falls } r < p+1-i \vee r > n-i \\ z_r^{(k)} z_r^{(k)} - z_r^{(k-1)} z_r^{(k-1)} & \text{falls } p+1-i \leq r \leq n-i \end{cases}$$

Inkrementelle Berechnung (IMR-IC)

Fall: $1 \leq i \leq p, 1 \leq j \leq p, i < j$

$$a_{ij}^{(k)} = a_{ji}^{(k)} = a_{ij}^{(k-1)} + (z_r^{(k)} - z_r^{(k-1)}) \times$$

$$\begin{cases} 0 & \text{falls } r < p + 1 - j \vee r > n - i \\ z_{r+j-i}^{(k-1)} & \text{falls } p + 1 - j \leq r < p + 1 - i \\ z_{r-j+i}^{(k-1)} & \text{falls } n - j < r \leq n - i \\ (z_{r+j-i}^{(k-1)} + z_{r-j+i}^{(k-1)}) & \text{falls } p + 1 - i \leq r \leq n - i \end{cases}$$

Inkrementelle Berechnung (IMR-IC)

Fall: $1 \leq i \leq p$

$$b_i^{(k)} = b_i^{(k-1)} + (z_r^{(k)} - z_r^{(k-1)}) \times$$

$$\begin{cases} 0 & \text{falls } r < p+1-i \vee r > n-i \\ z_{r+i}^{(k-1)} & \text{falls } p+1-i \leq r < p+1 \\ z_{r-i}^{(k-1)} & \text{falls } r > n-i \\ (z_{r+i}^{(k-1)} + z_{r-i}^{(k-1)}) & \text{falls } p+1 \leq r \leq n-i \end{cases}$$

Rekursive Algorithmus

-
-
- 1: **Eingabe:** Messung x , Reparatur/Label y
 - 2: **Ausgabe:** $\phi^{(k)}$
 - 3: **if** $k = 0$ **then**
 - 4: Init $A^{(0)}, B^{(0)}$ mit $Z^{(0)}, V^{(0)}$
 - 5: **else**
 - 6: r Index $y_r^{(k)} \neq y_r^{(k-1)}$
 - 7: Erstelle $A^{(k)}, B^{(k)}$ mit hilfe von $A^{(k-1)}$ und $B^{(k-1)}$ nach rekursive Formeln
 - 8: **end if**
 - 9: $\phi^{(k)} \leftarrow (A^{(k)})^{-1} B^{(k)}$
 - 10: **return** $\phi^{(k)}$
-

Überblick

- 1 Einführung
- 2 Grundlagen
- 3 Iterative Minimum Repairing
- 4 Evaluierung**
 - **Versuchsbeschreibung**
 - Ordnung
 - Schwellenwert
 - Maximale Anzahl von Iterationen
 - Markierungsrate

Versuchsbeschreibung

Versuchsaufbau

- Versuchsperson bewegt sich mit dem Handy auf den Hauptcampus
- Strecke ist festgelegt (x^{truth} , $x^{\text{truth}*}$)
- 186 von 742 GPS-Daten wurden als fehlerhaft festgestellt



Überblick

- 1 Einführung
- 2 Grundlagen
- 3 Iterative Minimum Repairing
- 4 Evaluierung**
 - Versuchsbeschreibung
 - **Ordnung**
 - Schwellenwert
 - Maximale Anzahl von Iterationen
 - Markierungsrate

Ordnung

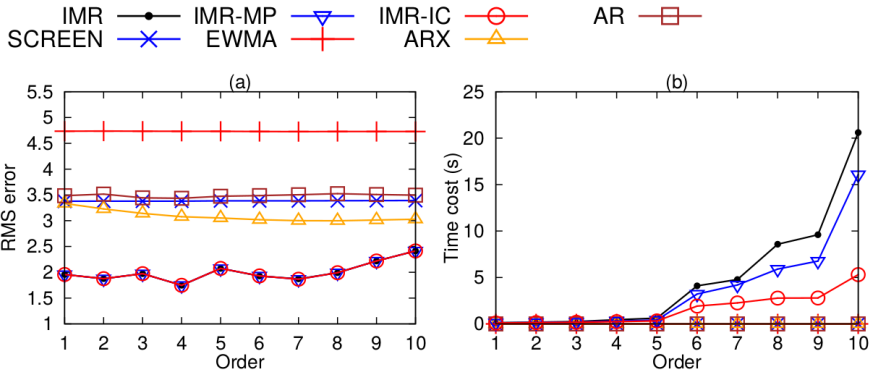


Abbildung: Unterschiedliche Ordnung p über GPS-Daten mit $\tau = 0.2$, Datengröße 750 und Markierungsrate 0.2

Überblick

- 1 Einführung
- 2 Grundlagen
- 3 Iterative Minimum Repairing
- 4 Evaluierung**
 - Versuchsbeschreibung
 - Ordnung
 - Schwellenwert**
 - Maximale Anzahl von Iterationen
 - Markierungsrate

Schwellenwert

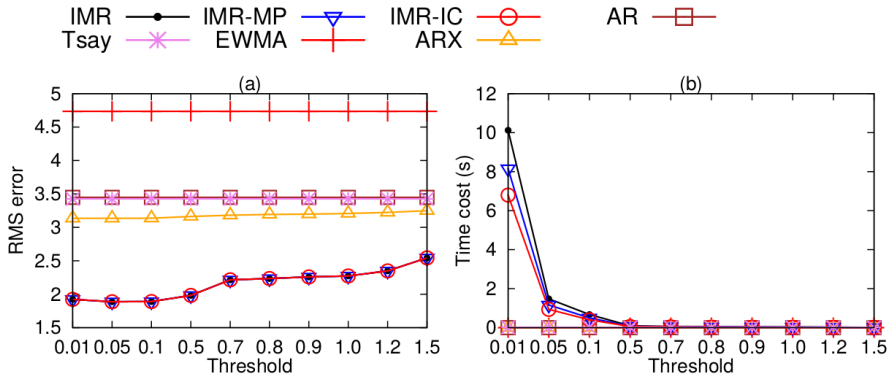


Abbildung: Unterschiedliche Schwellenwerte τ über GPS-Daten mit $p = 3$, Datengröße 750 und Markierungsrate 0.2

Überblick

- 1 Einführung
- 2 Grundlagen
- 3 Iterative Minimum Repairing
- 4 Evaluierung**
 - Versuchsbeschreibung
 - Ordnung
 - Schwellenwert
 - Maximale Anzahl von Iterationen**
 - Markierungsrate

Maximale Anzahl von Iterationen

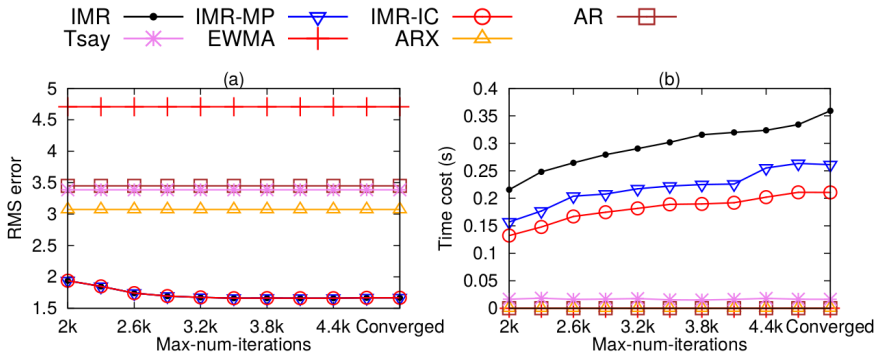


Abbildung: Unterschiedliche maximale Anzahl von Iterationen über GPS-Daten mit $\tau = 0, 2$, $p = 3$ und Datengröße 750

Überblick

- 1 Einführung
- 2 Grundlagen
- 3 Iterative Minimum Repairing
- 4 Evaluierung**
 - Versuchsbeschreibung
 - Ordnung
 - Schwellenwert
 - Maximale Anzahl von Iterationen
 - **Markierungsrate**

Markierungsrate

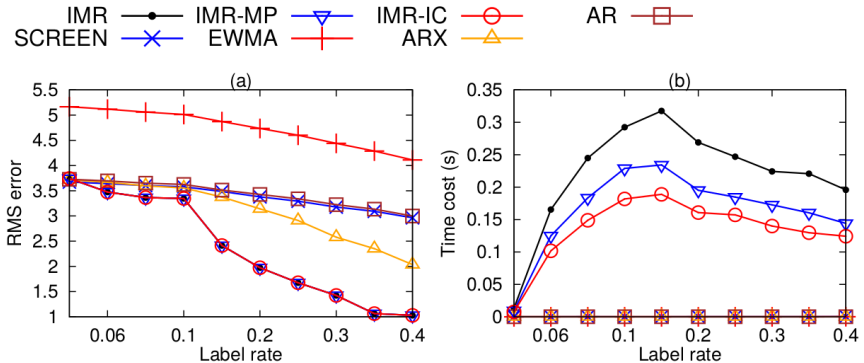


Abbildung: Unterschiedliche Markierungsraten über GPS-Daten mit $\tau = 0, 2$, $p = 3$ und Datengröße 750

Überblick

- 1 Einführung
- 2 Grundlagen
- 3 Iterative Minimum Repairing
- 4 Evaluierung
- 5 Schluss**
 - Zusammenfassung und Ausblick
 - Literatur

Zusammenfassung und Ausblick

Zusammenfassung

- Inkrementelle Reparatur der Daten durch Anomalienerkennungsverfahren
- Minimum-Change-Prinzip und temporäre Eigenschaften
- Bessere Laufzeit durch Matrix-Pruning
- IC Parameter Schätzung in $O(1)$
- Höhere Genauigkeit der Reparatur als State-of-the-art Verfahren

Zusammenfassung und Ausblick

Zusammenfassung

- Inkrementelle Reparatur der Daten durch Anomalienerkennungsverfahren
- Minimum-Change-Prinzip und temporäre Eigenschaften
- Bessere Laufzeit durch Matrix-Pruning
- IC Parameter Schätzung in $O(1)$
- Höhere Genauigkeit der Reparatur als State-of-the-art Verfahren

Ausblick

- Andere Modelle anstatt ARX

Vielen Dank
für die Aufmerksamkeit

Überblick

- 1 Einführung
- 2 Grundlagen
- 3 Iterative Minimum Repairing
- 4 Evaluierung
- 5 **Schluss**
 - Zusammenfassung und Ausblick
 - **Literatur**

Literatur I



Shaoxu song - tsinghua university.



Aoqian Zhang, Shaoxu Song, Jianmin Wang, and Philip S Yu.

Time series data cleaning: From anomaly detection to anomaly repairing.

Proceedings of the VLDB Endowment, 10(10):1046–1057, 2017.