Kapitel 4
Ähnlichkeitssuche in Zeitreihen

4 Ähnlichkeitssuche in Zeitreihen

Überblick

4.1 Einleitung

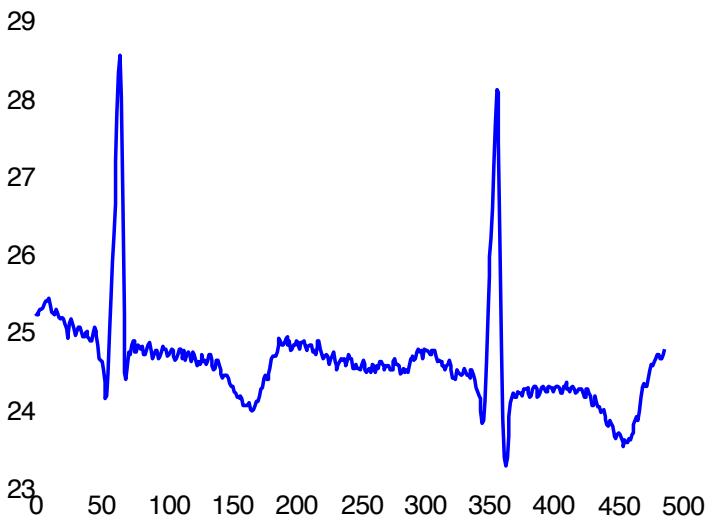
4.2 Matching-basierte Analyse

4.3 Threshold-basierte Analyse

4.1 Einführung

4.1 Einführung

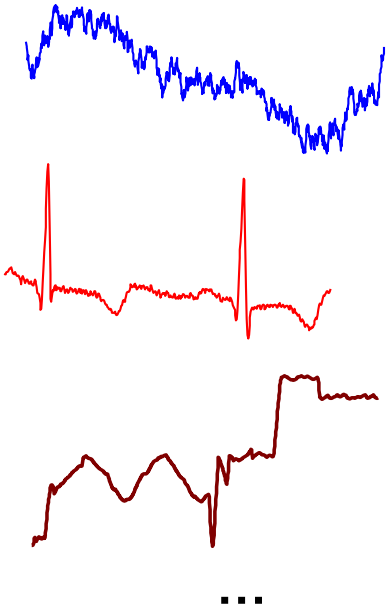
- Zeitreihe = Sammlung von Beobachtungen die zeitlich sequentiell gemacht werden/wurden
  - Zeitreihe $o \in (\Re^d \times Time)$ mit $o = [(o_1,t_1), \ldots,(o_n,t_n)]$
    - Meist $d = 1$
    - *Time* ist die Zeitdomäne, meist diskret
  - Diskrete Zeitreihe daher $o = [o_1, \ldots, o_n]$
    - Oft werden die Werte zwischen zwei Zeitpunkten interpoliert



4.1 Einführung

- Zeitreihen sind allgegenwärtig
  - Menschen messen alles Mögliche ...
    - Die Zustimmung der Bevölkerung zur Regierung
    - Den Blutdruck
    - Die Anzahl der Sonnenstunden in Freiburg pro Jahr
    - Der Wert der BVB-Aktie
    - Die Anzahl der Webhits pro Sekunde

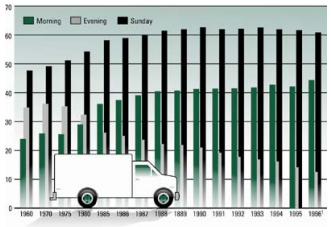
    … und diese Dinge verändern sich über die Zeit hinweg

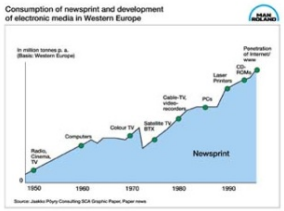


...

  - Zeitreihen werden in allen Anwendungsgebieten erzeugt

Naturwissenschaft Medizin Wirtschaft





...

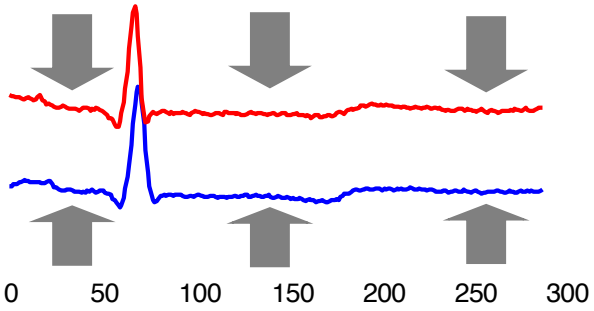
4.1 Einführung

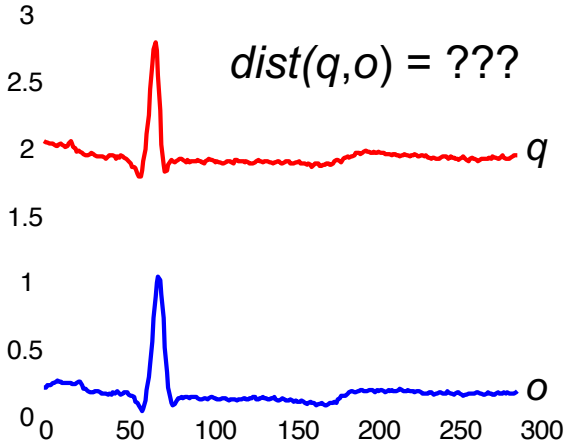
- Herausforderungen
  - Große Datenvolumina
    - 1h EKG: ca. 1GB
    - Typischer Weblog: ca. 5GB pro Woche
    - Datenbank mit Space Shuttle Messwerten: ca. 160 GB
    - …
  - Heterogene Daten
    - Verschiedene Datenformate
    - Verschiedene Sampling Raten
    - Verrauschte Signale
    - Fehlende Werte
    - …
  - Subjektive Wahrnehmung der Ähnlichkeit abhängig von
    - User
    - Anwendung
    - Analyse-Art (Matching, Threshold-basiert, …)

4.1 Einführung

- Daten Vorverarbeitung
  - Beseitigung von Verzerrungen in den Rohdaten
  - Die wichtigsten Verzerrungen
    - Offset Translation
      - Ähnlich Zeitreihen mit unterschiedlichen Offsets
      - Verschiebung aller Zeitreihen um den Mittelwert *MW*:

$$\forall 1 \le i \le |o| : o_i = o_i - \mathrm{MW}(o)$$

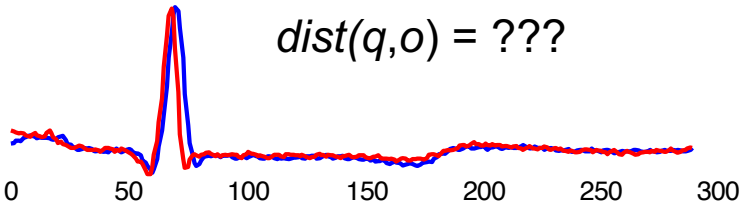




$q = q - MW(q)$

$o = o - MW(o)$

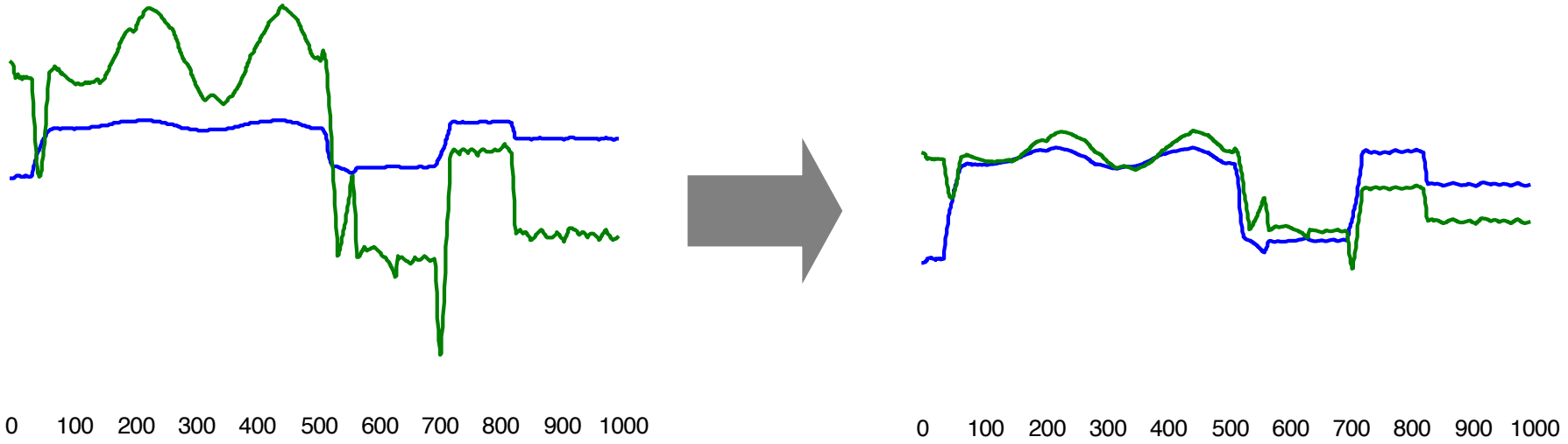
$dist(q,o) = ???$



4.1 Einführung

- Amplituden Skalierung
  - Zeitreihen mit ähnlichem Verlauf aber unterschiedlichen Amplituden
  - Verschiebung der Zeitreihen um den Mittelwert (*MW*) und Normierung der Amplitude mittels der Standard Abweichung (StD):

$$\forall 1 \leq i \leq |o| : o_i = (o_i - \mathrm{MW}(o)) \,/\, \mathrm{StD}(o)$$



$$q = (q - MW(q)) \,/\, \mathrm{StD}(q)$$

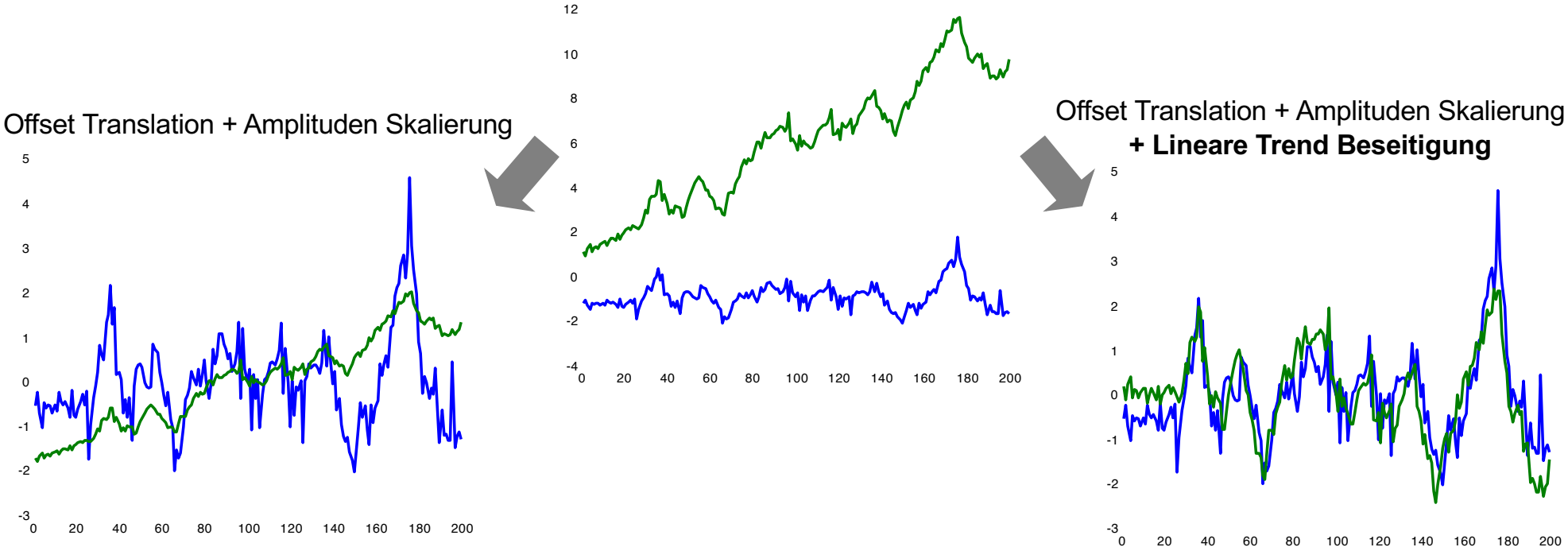
$$o = (o - MW(o)) \,/\, \mathrm{StD}(o)$$

4.1 Einführung

- Lineare Trends
  - Ähnliche Zeitreihen mit unterschiedlichen Trends
  - Intuition:

    - Bestimme Regressionslinie

    - Verschiebe Zeitreihe anhand dieser Linie



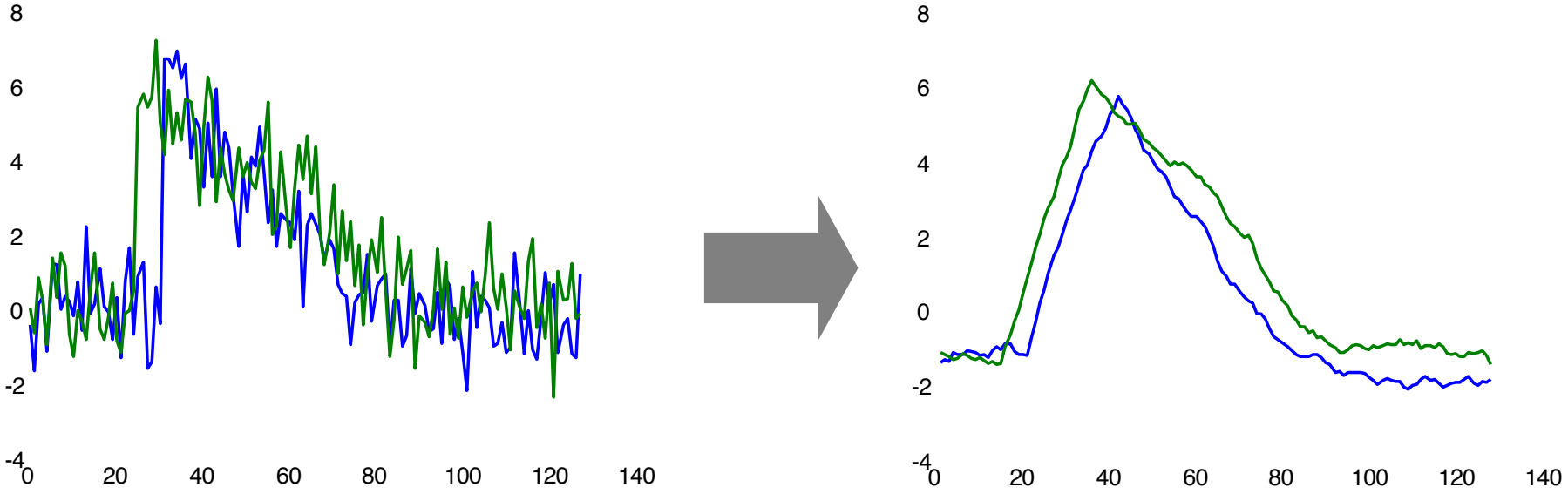
4.1 Einführung

- Bereinigung von Rauschen
  - Ähnliche Zeitreihen mit hohem Rauschanteil
  - Glättung:

    Bilde für jeden Wert o_i den Mittelwert über alle Werte $[o_{i-k}, \ldots, o_i, \ldots, o_{i+k}]$ für ein gegebenes k

    k bestimmt den Grad der Glättung,
    *Frequenzen f > 1/(2k*dt) werden weggefiltert, d.h. Tiefpassfilter).*

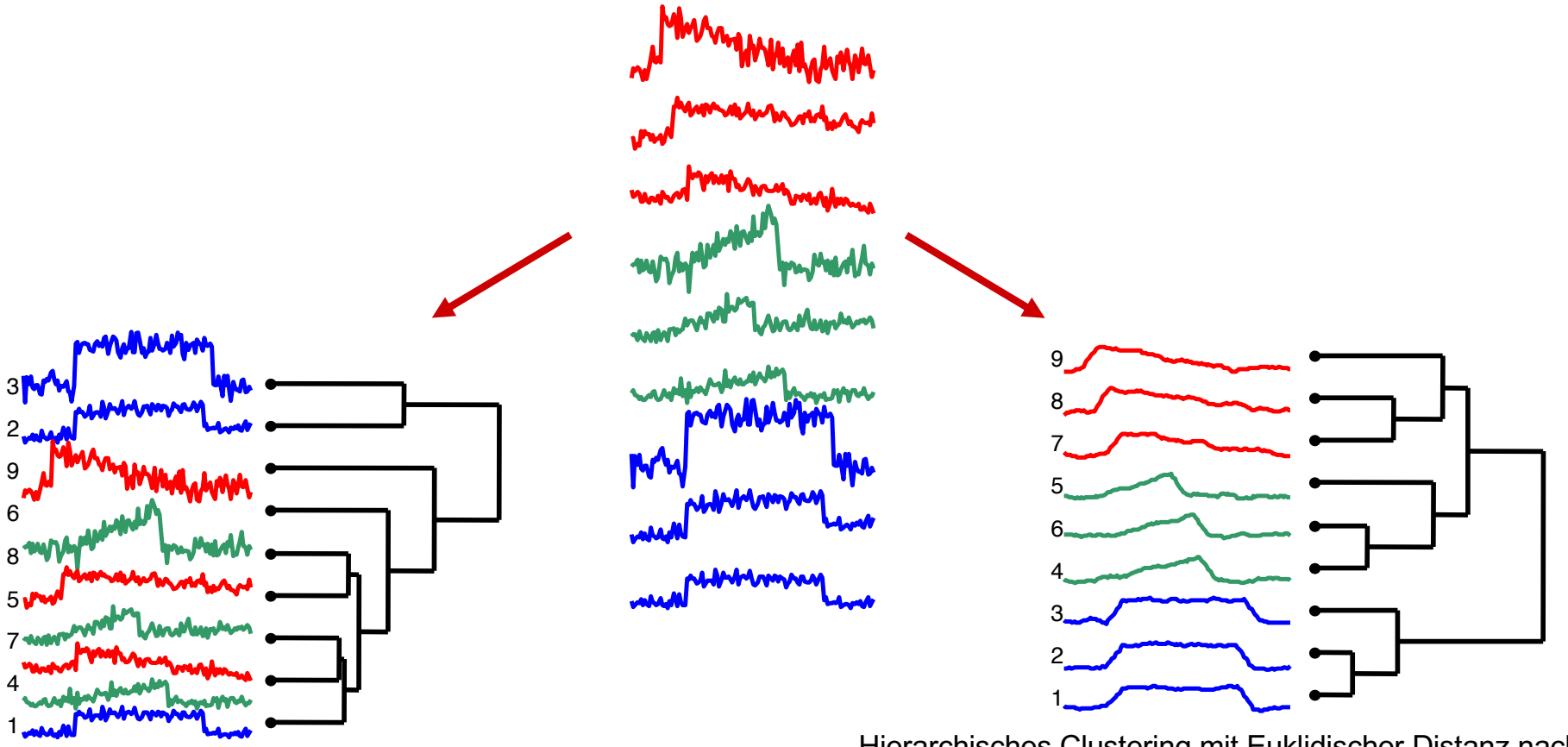
    dt = Zeit zwischen zwei aufeinanderfolgenden Messwerten o_i und o_{i+1}



4.1 Einführung

- Beispiel

Rohdaten: Zeitreihen aus drei Klassen (erkennbar durch unterschiedliche Farbcodierung)



Hierarchisches Clustering der Rohdaten mit Euklidischer Distanz

Hierarchisches Clustering mit Euklidischer Distanz nach Bereinigung von Rauschen, Beseitigung linearer Trends, Amplituden Skalierung und Offset Translation

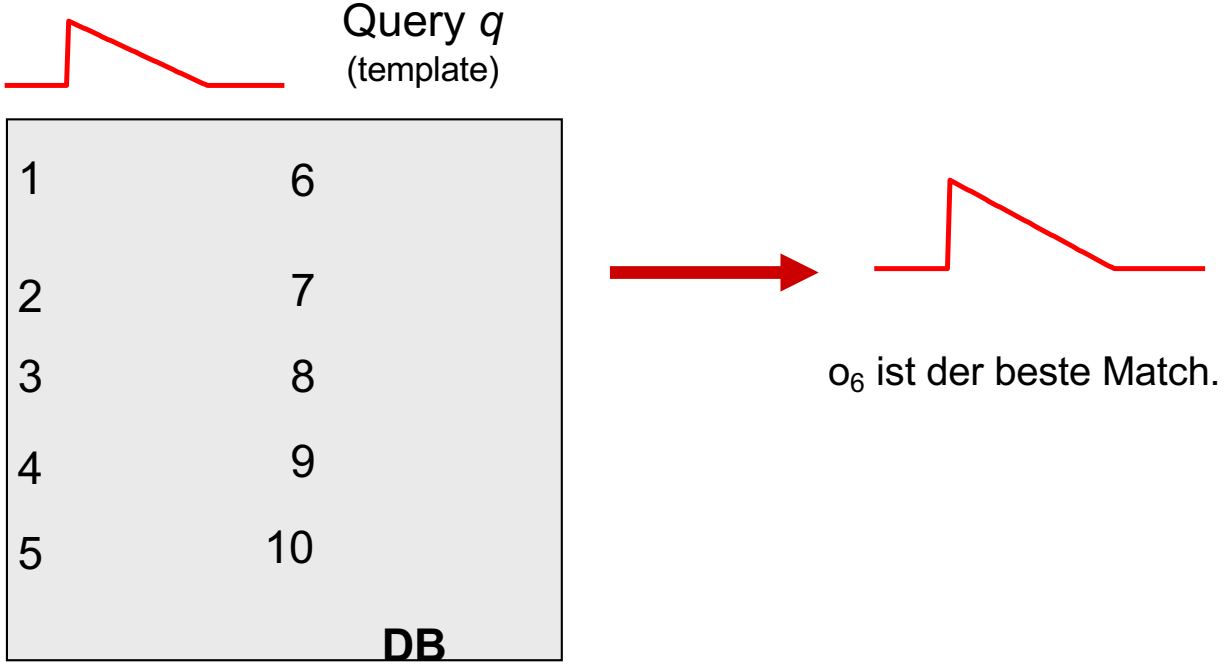
4.1 Einführung

- Zusammenfassung
  - Die Rohdaten können unterschiedlichste Verzerrungen haben, die vor der Analyse beseitigt werden sollten
  - Welche Vorverarbeitungsschritte ausgeführt werden sollen, hängt typischerweise von der Anwendung ab
  - **ACHTUNG**: Oft sind die Verzerrungen die interessanten Informationen, die man durch eine Analyse finden will
  - Verschiedene „high-level“ Repräsentationen von Zeitreihen, die wir später kennen lernen, bieten elegante Möglichkeiten, diese Verzerrungen in den Griff zu bekommen

4.2 Matching-basierte Analyse

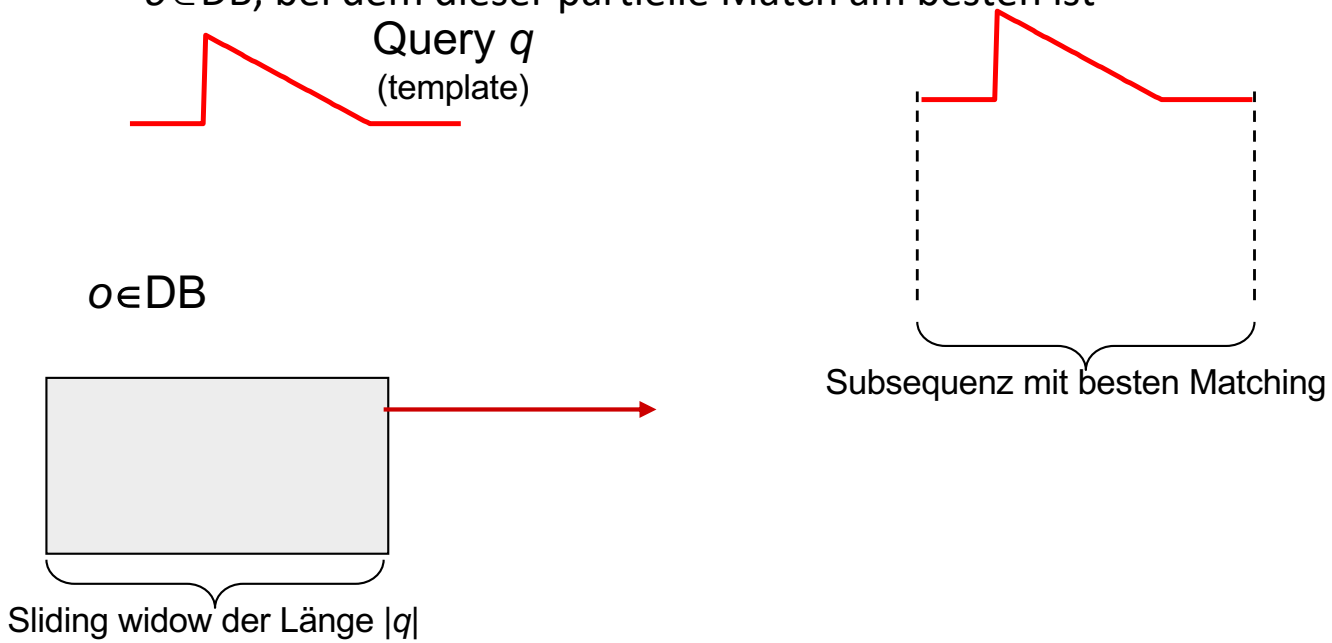
4.2 Matching-basierte Analyse

- Analyse-Arten
  - Gesamt-Suche („Whole-Matching“)
    - Gegeben: Query-Objekt *q* (Zeitreihe oder Template), Distanzfkt. *dist*
    - Gesucht: Zeitreihe $o \in DB$, die am besten mit *q* „matched“ (als Ganzes)



4.2 Matching-basierte Analyse

- Subsequenz-Suche („Subsequence Matching“)
  - Gegeben: Query-Objekt *q* (Zeitreihe oder Template), Distanzfkt. *dist*
  - Gesucht: Zeitabschnitt, bei dem $o \in DB$ am besten mit *q* „matched“, bzw. dasjenige $o \in DB$, bei dem dieser partielle Match am besten ist

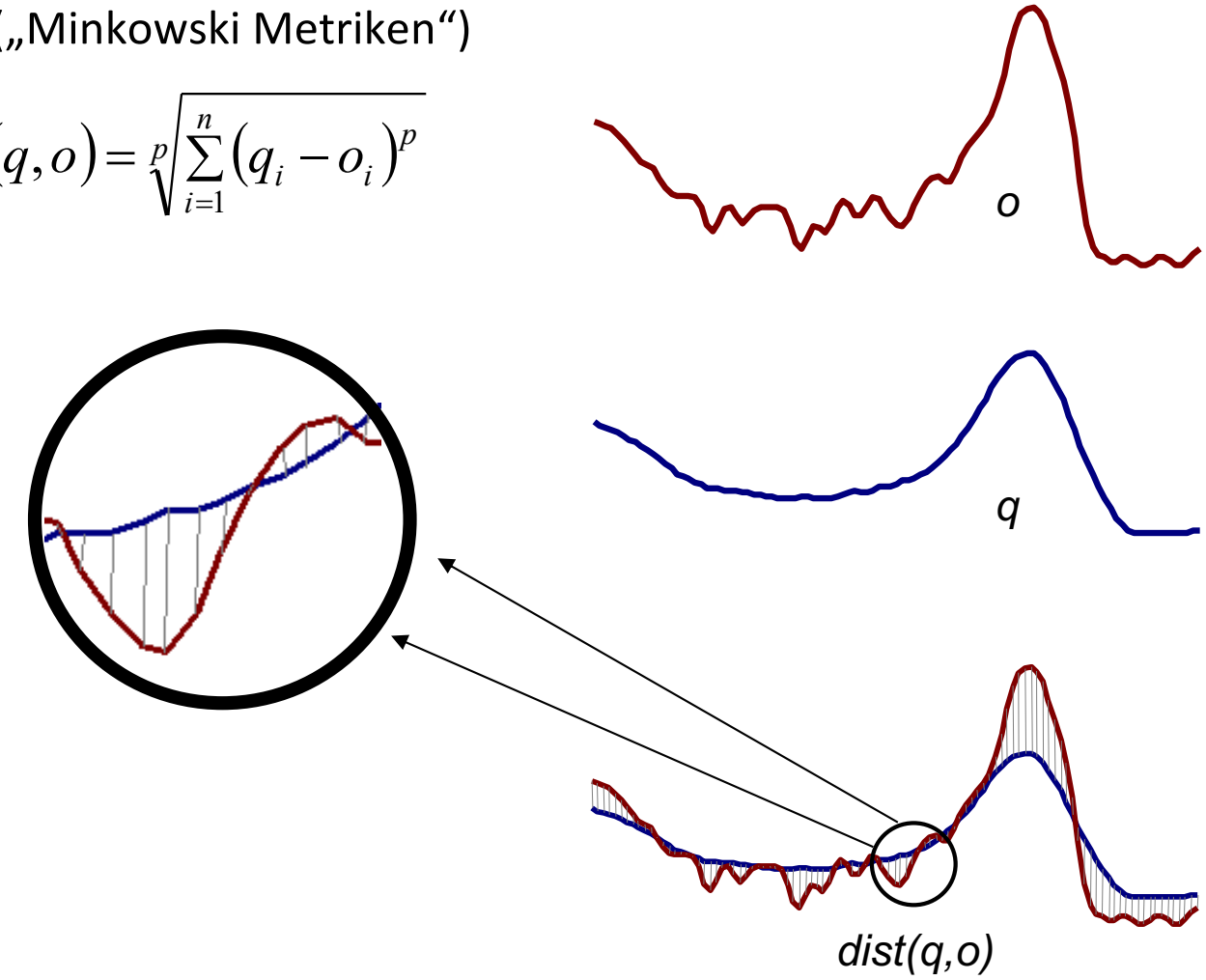


  - Bemerkung:
    - Ist die Länge der Query-Subsequenz bekannt und fix, kann die Subsequenz-Suche immer in eine Gesamt-Suche überführt werden
    - Wie? Verschiebe ein Fenster („sliding window“) über die Zeitreihen und materialisiere die Inhalte und organisiere diese in einem Index (z.B. R-Baum)

4.2 Matching-basierte Analyse

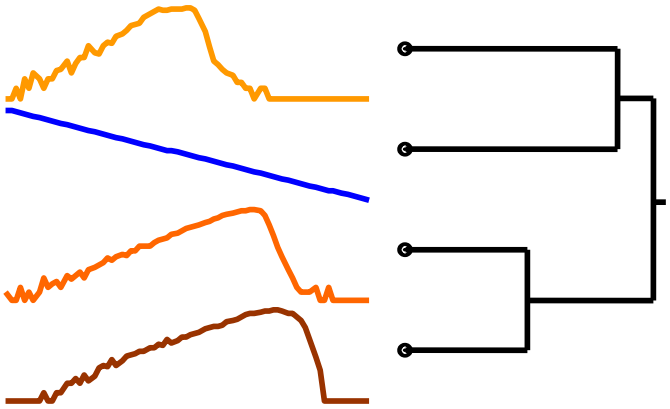
- Ähnlichkeit von Zeitreihen
  - Lp-Normen („Minkowski Metriken“)

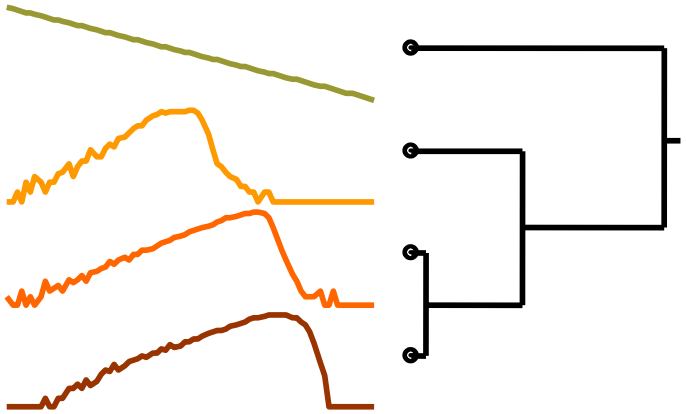
$$dist(q,o) = \sqrt[p]{\sum_{i=1}^{n} (q_i - o_i)^p}$$

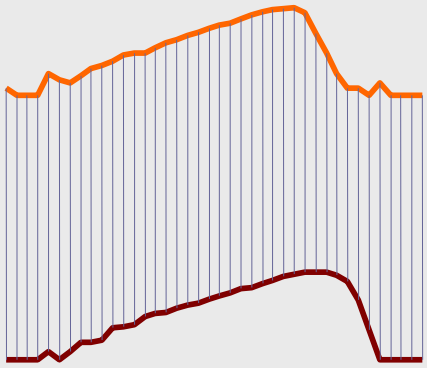


4.2 Matching-basierte Analyse

- Dynamic Time Warping (DTW)

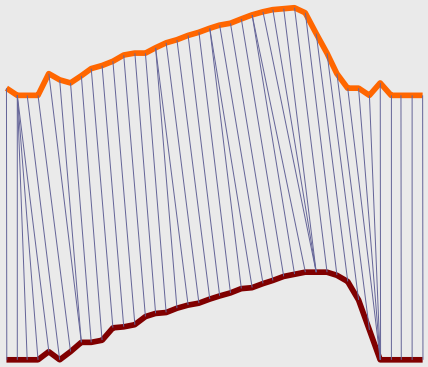






Fixierte Zeitachse (Lp-Norm)

Sequenzen werden „Eins-zu-Eins“ angeglichen und verglichen (Alignment)

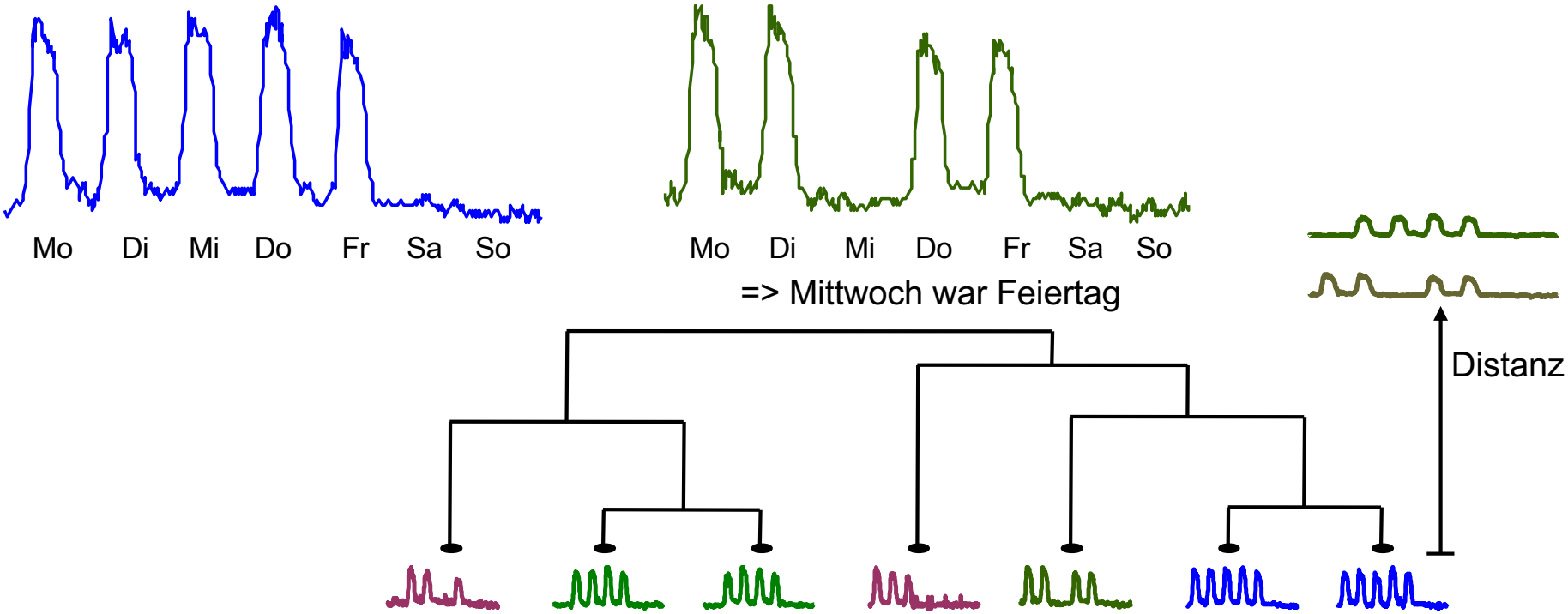


Verzogene („Gewarped”) Zeitachse

Nichtlineare Alignments sind möglich

4.2 Matching-basierte Analyse

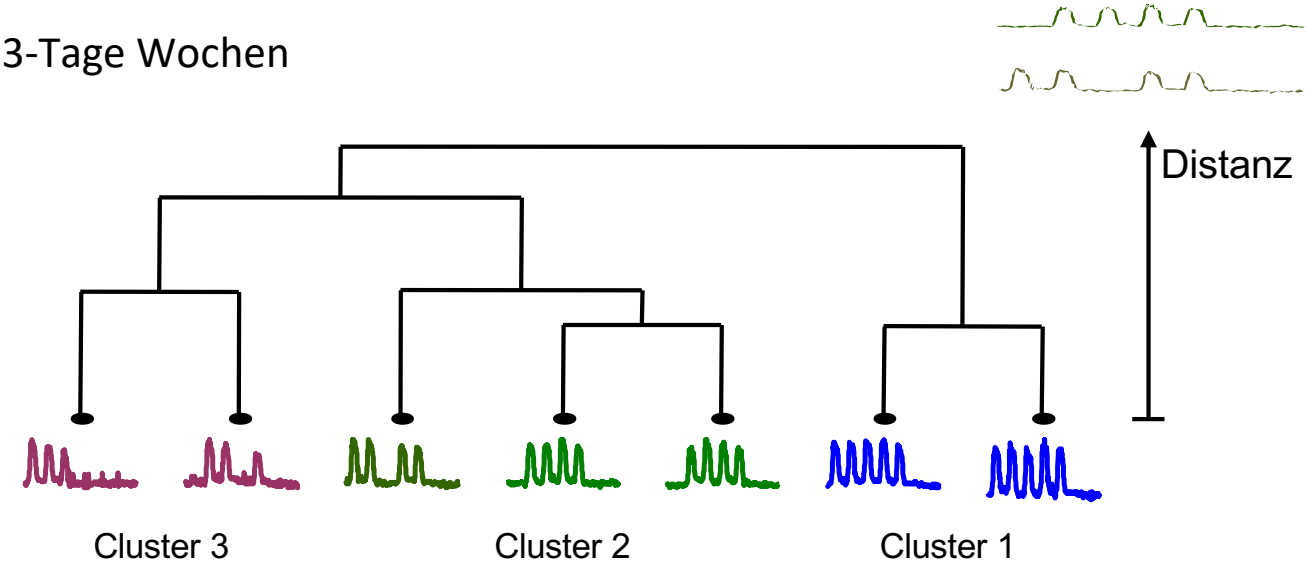
- Vergleich
  - DB enthält Zeitreihen, die den wöchentlichen Bedarf einer Firma messen (1 Zeitreihe entspricht 1 Woche)



  - Ergebnis eines Clusterings mit Euklidischer Distanz
    - Die beiden 5-Tage Wochen korrekt gruppiert
    - Die drei 4-Tage Wochen und die beiden 3-Tage Wochen vermischt

4.2 Matching-basierte Analyse

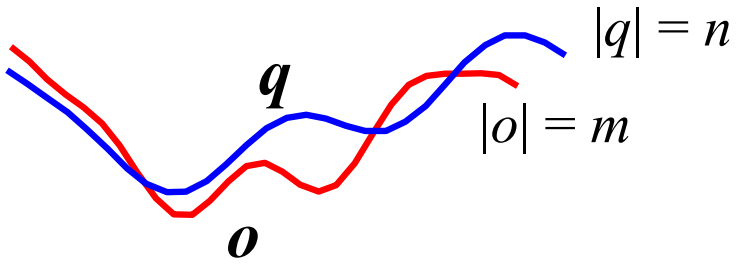
- Ergebnis eines Clusterings mit Dynamic Time Warping
  - Cluster 1: 5-Tage Wochen
  - Cluster 2: 4-Tage Wochen
  - Cluster 3: 3-Tage Wochen

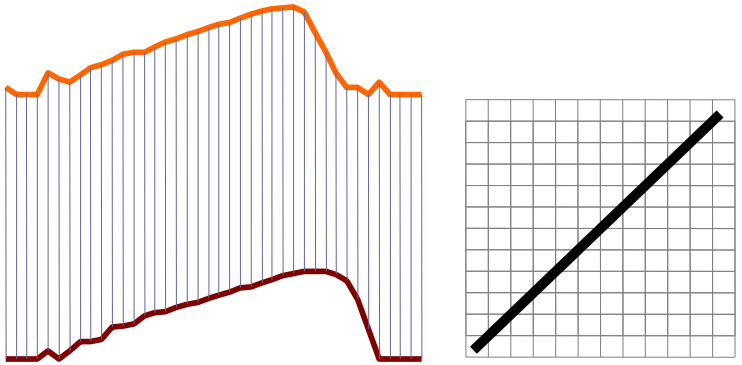


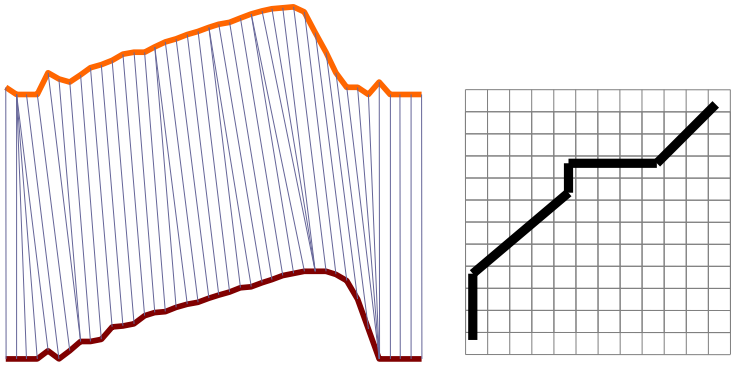
- Laufzeit:
  - Euklidische Distanz: ca. 1 Sekunde
  - DTW: ca. 3,5 Stunden

4.2 Matching-basierte Analyse

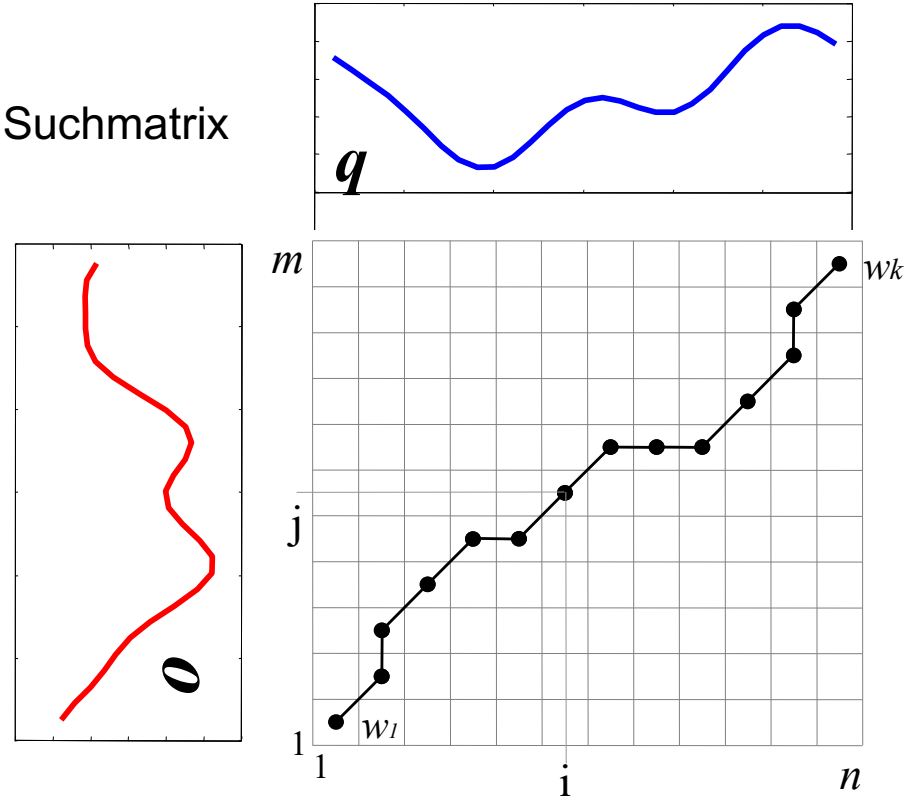
- Berechnung der DTW Distanz
  - Gegeben: Zeitreihen *q* und *o* unterschiedlicher Länge
  - Finde mapping von allen q_i auf o_j mit minimalen Kosten







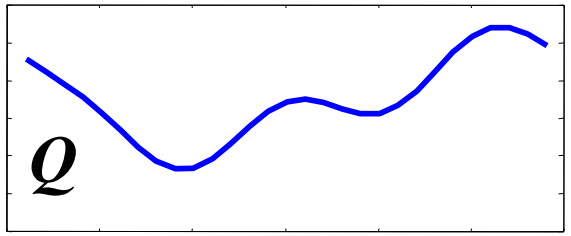
Suchmatrix



4.2 Matching-basierte Analyse

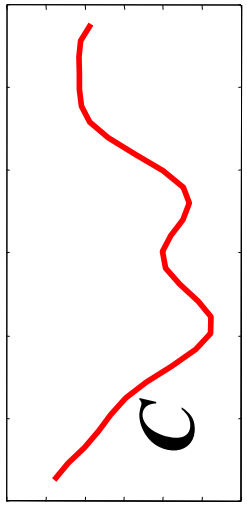
- Suchmatrix
  - Alle möglichen mappings von q auf o können als „warping“ Pfad in der Suchmatrix aufgefasst werden
  - Von all diesen Mappings suchen wir den Pfad mit den niedrigsten Kosten

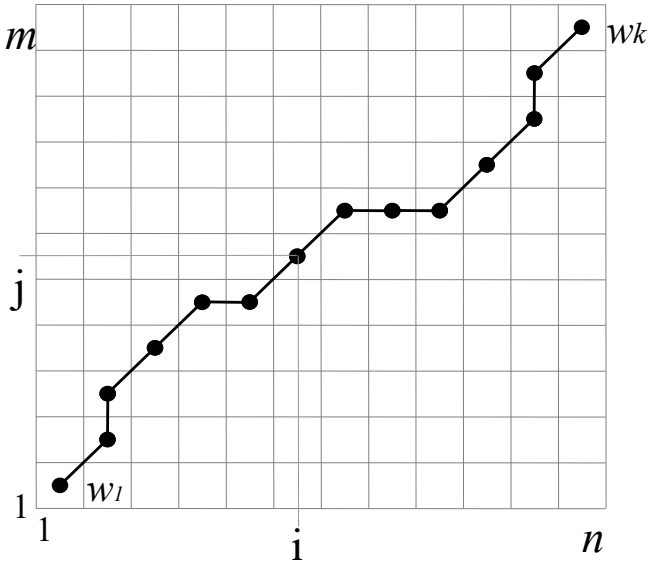
$$DTW(q,o) = \min\left\{\sqrt{\sum_{k=1}^{K} w_k}\Big/K\right\}$$



  - Theoretisch: exponentiell viele Pfade
  - Praktisch: Dynamisches Programmieren

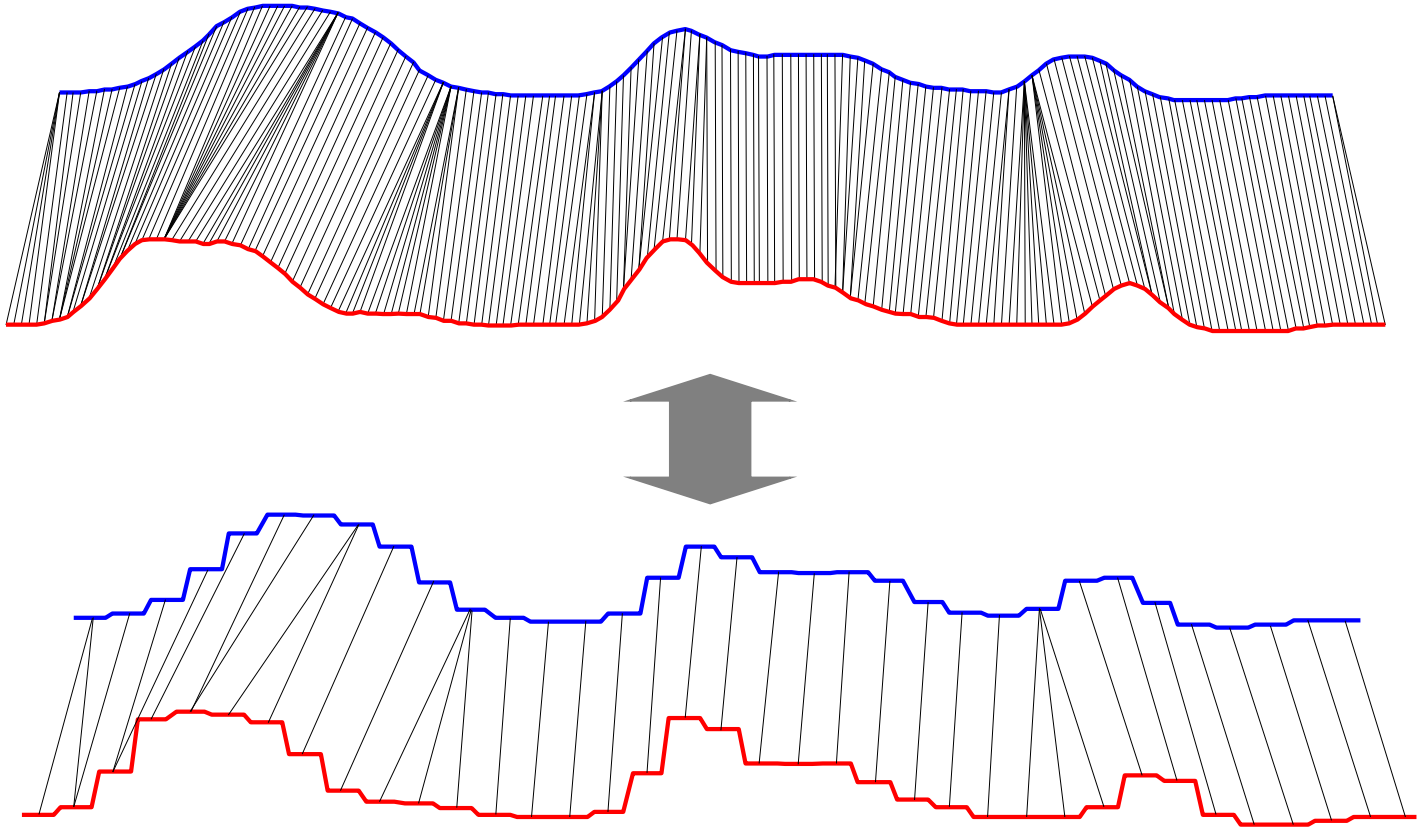
    => Laufzeit ($n \cdot m$)





4.2 Matching-basierte Analyse

- Approximative Dynamic Time Warping Distanz
  - Idee
    - Approximiere die Zeitreihen (komprimierte Repräsentation, Sampling, …)
    - Berechne DTW auf den Approximationen

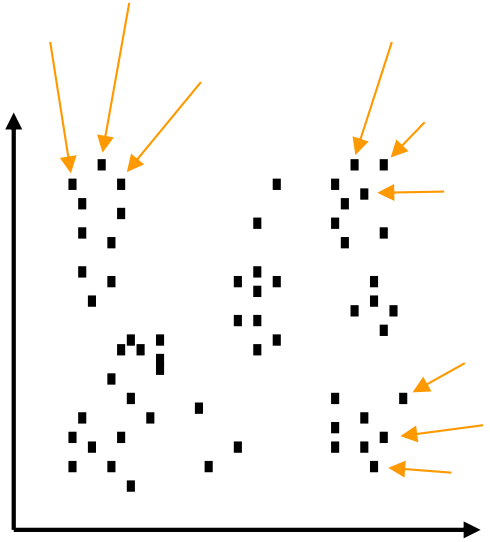


4.2 Matching-basierte Analyse

- Anfragebearbeitung
  - Indexierung:
    - eine Zeitreihe der Länge *n* kann als

      n-dimensionaler Feature-Vektor

      modelliert werden



  - Problem:
    - Zeitreihen meist sehr lang
    - Curse of Dimensionality!!!
  - Lösung: GEMINI-Framework [Faloutsos, Ranganathan, Maolopoulos. SIGMOD 1994]
    - Basiert auf Dimensionsreduktion
    - Transformiere *n*-dimensionale Zeitreihen in *d*-dimensionalen Feature-Vektor (*d*<<*n*)
    - Definiere eine Distanzfunktion für den *d*-dimensionalen Featurvektor, die die untere-Schranke-Eigenschaft erfüllt

$$dist_{reduziert}(p,q) \leq dist_{original}(p,q)$$

4.2 Matching-basierte Analyse

- Techniken zur Dimensionsreduktion für Zeitreihen (Überblick)