Vorlesung Maschinelles Lernen
LACE

Katharina Morik

LS 8 Künstliche Intelligenz Fakultät für Informatik
Technische Universität Dortmund

13.1.2009

Gliederung

1. Organisation von Sammlungen
   - Web 2.0
   - Clustering verteilter Daten
2. LACE
3. Experimente mit LACE
4. Musik als Daten
   - Lernende, adaptive Merkmalsextraktion
   - Merkmalsübertragung

Organisation von Sammlungen

Sammlungen von Fotos, Musik, Filmen bevölkern PCs und das Internet. Sie sind organisiert

- in Taxonomien nach vorgegebenen Kriterien
  - iTunes: Genre, Artist, Album, Jahr
- in Taxonomien nach eigenen Kriterien
  - flickR: Sammlung, Album, Gruppen – annotiert wird mit eigenen tags.
- einfache Dateien, evtl. mit Benutzeroberfläche
  - iPhoto: Ereignisse, jedes Bild kann annotiert werden.

Wie organisieren Menschen Medien?

- Studie von Jones, Cunningham, Jones (2004): Studenten wurden befragt, wie sie ihre CDs, DVDs, Bücher organisieren.
  - Nachttisch, spezieller Schrank, Auto, Küche
  - Gelegenheiten zur Nutzung
  - Aktualität, Anschaffungszeitpunkt
- Studie von Vignoli (2004): Ordnung digitaler Musik auf PCs wurden untersucht.
  - Meist wurden hierarchische Strukturen aufgebaut.
  - Es gibt immer einen Ordner mit nicht einsortierter Musik.
- Studie PG 461 "Kollaboratives Strukturieren von Multimediadaten für Peer-to-Peer-Netze"
  - Verschiedene Aspekte: Gelegenheiten ("beim Autofahren", "Dinner", ”Party"), Personen (”für Susie"), Erinnerungen (”Sommer03"), Stimmungen, Tempi, Genres
  - Wieder gibt es Ordner mit nicht einsortierter Musik.

Automatisches Sortieren von Mediensammlungen

- Medien sollen hierarchisch strukturiert werden.
- Die Taxonomien sollen personalisiert sein.
  - Die Bezeichner sind unterschiedlich: was dem einen ”fröhliche Tanzmusik”, gehört bei dem anderen unter ”Depression” (The Cure).
  - Bereiche, die einer fein strukturiert, fasst der andere zusammen.
  - Verschiedene Benutzer stellen verschiedene Mengen als ähnlich betrachteter Medien zusammen.
- Derselbe Benutzer verwendet mehrere, unterschiedliche Hierarchien (Aspekte), die teilweise gleiche Medien abdecken.
- Die Einsortierung neuer Medien soll automatisch erfolgen.
- Die Struktur soll automatisch erweitert werden, ohne den Benutzer zur bevormunden.

Web 2.0

- Semantic Web:
  - Semantische Beschreibung
  - Vorgegebene, allgemeine Ontologie
  - Logische Beschreibungssprache
  - top-down Modellierung
- Web 2.0
  - Freies Tagging der Benutzer
  - Entstehende Folksonomies
  - Statistische Methoden
  - Empfehlungssysteme

Sammlungen im Web 2.0

- Verschiedene Benutzer laden ihre Medien hoch.
- Verschiedene Benutzer annotieren ihre Medien.
- Kollaborative Empfehlung:
  - Ein Benutzer sind einander ähnlich, wenn sie ähnliche Mengen von Medien ausgewählt haben.
  - Medien sind einander ähnlich, wenn sie in Sammlungen ähnlicher Benutzer vorkommen.
  - Meist werden nur flache Medienmengen betrachtet (Amazon, Last.fm). Es werden auch nur Listen von Medien empfohlen.
- Für die automatische Unterstützung der Strukturierung reicht das nicht.

Clustering Mediensammlungen

- **Ziel:** Hierarchisches Clustering erzeugt für einen Benutzer anhand seiner und der Clusterings anderer Benutzer je Aspekt mehrere Taxonomien zur Auswahl.
  - Wie kann das Benutzer gegebene Clustering beibehalten und nur ergänzt werden? –> Supervised Clustering
  - Wie kann ein Benutzer von den Strukturierungen anderer Benutzer profitieren? –> Distributed Clustering, Ensemble Clustering
  - Wie kann das Verfahren mehrere alternative Clusterings zur Auswahl anbieten? –> Nonredundant Clustering

Supervised Clustering

- **Constraint Clustering** (Cohn, Caruana, McCallum 2003) beachtet bei der Optimierung vom Benutzer gegebene Nebenbedingungen
  - $must - link(\vec{x_g}, \vec{x'_g})$, d.h. $\vec{x_g}, \vec{x'_g}$ müssen im selben Cluster sein;
  - $cannot - link(\vec{x_g}, \vec{x_h})$, d.h. $\vec{x_g}, \vec{x_h}$ dürfen nicht im selben Cluster sein.
- **Supervised Clustering** (Finley, Joachims 2005) beachtet bei der Optimierung als Nebenbedingungen, dass einige Cluster mit zugeordneten Beobachtungen vorgegeben sind: $C(i) = k$ für
  $\vec{x_i}$, i=1,...,M, M<N
  $C_k, k = 1, ..., L, L \leq K$
- Leider nur für flache Clusterings und nicht für mehrere, verteilte gegebene Clusterings!

Distributed Clustering

- Verteilte Daten sollen gruppiert werden.
- Horizontale Verteilung:
  - Alle Daten haben die selben Merkmale, sind aber auf verschiedene Rechner verteilt.
  - Kein Datum ist mehr als einem Rechner zugeordnet.
  - Typisches Beispiel: Filialen eines Geschäfts.
- Vertikale Verteilung:
  - Daten der verschiedenen Rechner haben unterschiedliche Merkmale.
  - Das selbe Objekt ist auf mehreren Rechnern zu finden.
  - Typisches Beispiel: Mediensammlungen Web 2.0.
- Ziel ist ein Konsens-Modell als gemeinsames Clustering für alle Daten.
- Das ist nicht das Ziel bei der Strukturierung persönlicher Mediensammlungen!

Ensemble Clustering

- Ensemble Clustering kombiniert eine Menge gegebener Clusterings (Strehl, Ghosh 2002).
- Alle Clusterings decken die selbe Menge von Beobachtungen ab.
  - Zusätzliches Ähnlichkeitsmaß: kommen gemeinsam in einem Cluster vor (Topchy, Jain, Punch 2003);
  - Zuordnung zu einem gegebenen Cluster als zusätzliches Merkmal einer Beobachtung – dann in diesem Raum k-Means anwenden!
- Wieder wird ein Konsens-Modell erzeugt!

Nonredundant Clustering

- Gegeben ein Clustering $C(i) = k$ für Beobachtungen $\vec{x_i}, i = 1, ..., N$ und Cluster $C_k, k = 1, ..., K$
- finde ein alternatives Clustering C', das möglichst orthogonal zu C ist. (Gondek, Hofmann 2004)
- Das Verfahren erhält keine gegebenen Strukturierungen, sondern bietet Alternativen zum gesamten Clustering an.

Es gibt noch kein geeignetes Verfahren für das Strukturieren persönlicher Sammlungen im Web 2.0

- Bisherige Ansätze reichen nicht aus:
  - Supervised clustering ist noch nicht geeignet für hierarchische Strukturen und die Eingabe mehrerer Clusterings.
  - Distributed clustering und Ensemble Clustering erstellen ein Konsens-Modell, das die eigene Annotation von Benutzern überschreiben würde.
  - Nonredundant clustering erhält in den Alternativen nicht das gegebene Clustering.
- Wir mussten also ein eigenes Verfahren entwickeln: Localized Alternative Clustering of Ensembles

Lernaufgabe Localized Alternative Clustering of Ensembles

- Wir sprechen jetzt statt von der Zuordnung $C(i) = k$ einer Beobachtung $\vec{x_i}$ zu einem Cluster C_k von dem Clustering φ_i von einer Menge von Beobachtungen S_i auf ein Cluster G_i.
- Gegeben eine Menge $S \subseteq X$, eine Menge von Clusterings $I \subseteq \{\varphi_i : S_i \to G_i\}$ und eine Qualitätsfunktion

$$q : 2^{\Phi} \times 2^{\Phi} \times 2^{S} \to \mathcal{R} \qquad (1)$$

localized alternative clustering ensembles findet Clusterings $O \subseteq \{\varphi_i | \varphi_i : S_i \to G_i\}$ so dass die Qualität $q(I, O, S)$ maximiert wird und für jedes $\varphi_i \in O$ gilt, dass S Teil seines Ursprungsbereichs ist: $S \subseteq D_{\varphi_i}$.

φ als hierarchisches Clustering

- Die Cluster sollen nicht auf einer Ebene liegen, sondern eine Taxonomie bilden.
- Die unterste Ebene enthält Mengen von Beobachtungen.
- Jede Ebene enthält Cluster, die die Cluster der Ebene darunter subsummieren: jeder Teilbaum von Clustern ist eine Taxonomie.
- Die oberste Ebene enthält ein Cluster mit allen Beobachtungen.
- Man unterscheidet ein Vorgehen bottom-up (agglomerativ) und top-down (aufteilend).
- $\varphi_i : S_i \to G_i$ soll die Menge S_i hierarchisch aufteilen, d.h. G_i soll eine Hierarchie von Clustern sein.

Zur Erinnerung: Agglomeratives Clustering

- Stufenweise werden Beobachtungen zu übergeordneten Clustern verschmolzen.
- Grundlage ist die Unähnlichkeit von Clustern: solche mit geringster Unähnlichkeit werden verschmolzen.
- Die Unähnlichkeit $d(G, H)$ der Cluster G, H wird berechnet durch den Abstand $d_{gh} = D(\vec{x_g}, \vec{x_h})$, wobei $\vec{x_g} \in G, \vec{x_h} \in H$.
- Welche Beobachtungen genutzt werden, macht den Unterschied zwischen den 3 wichtigsten Maßen zur Cluster-Unähnlichkeiten aus.
  - Single Linkage Clustering: Die Unähnlichkeit zwischen Cluster G und H ist die Unähnlichkeit der nächsten Punkte.
  - Complete Linkage Clustering: Die Unähnlichkeit zwischen Cluster G und H ist die Unähnlichkeit der entferntesten Punkte.
  - Average Linkage Clustering: Die Unähnlichkeit zwischen Cluster G und H ist die durchschnittliche Unähnlichkeit aller Punkte in G von allen in H.

Erweiterung eines Clustering

Wir wollen ein gegebenes Clustering erweitern, d.h.:

- Bestehende Zuordnungen bleiben.
- Bisher abgedeckte Beobachtungen bleiben abgedeckt.
- Zusätzliche Beobachtungen werden abgedeckt.

Erweiterte Funktion

$\varphi'_i : S'_i \to G_i$ ist die erweiterte Funktion für $\varphi_i : S_i \to G_i$, wenn $S_i \subset S'_i$ und $\forall \vec{x} \in S_i : \varphi_i(\vec{x}) = \varphi'_i(\vec{x})$.

Beutel von Clusterings

Wir wollen die noch nicht strukturierten Beobachtungen in S durch vorhandene Clusterings $\varphi_1, ..., \varphi_m$ abdecken.

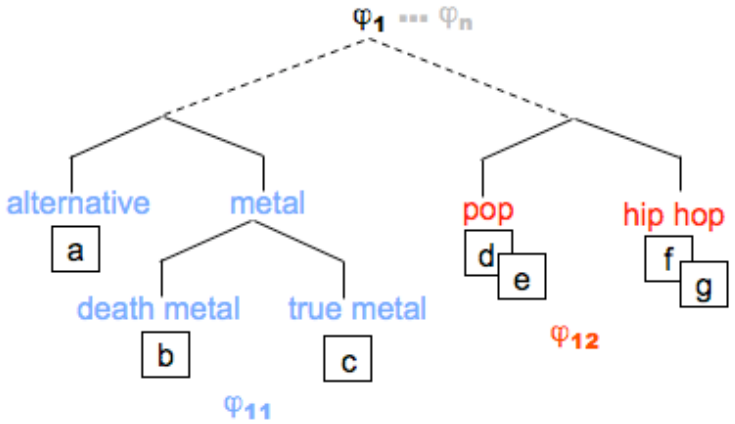
Beutel von Clusterings

Sei I eine Menge von Clusterings. Ein Beutel von Clusterings ist eine Funktion

$$\varphi_i(\vec{x}) = \begin{cases} \varphi'_{i1}(x), & \text{wenn } \vec{x} \in S'_{i1} \\ \vdots & \vdots \\ \varphi'_{ij}(x), & \text{wenn } \vec{x} \in S'_{ij} \\ \vdots & \vdots \\ \varphi'_{im}(x), & \text{wenn } \vec{x} \in S'_{im} \end{cases} \quad (2)$$

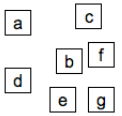
wobei jedes φ'_{ij} eine Erweiterung eines $\varphi_{ij} \in I$ ist und $\{S'_{i1}, \ldots, S'_{im}\}$ ist eine Partitionierung von S.

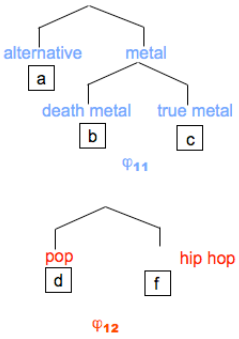
Beutel von Clusterings im Bild



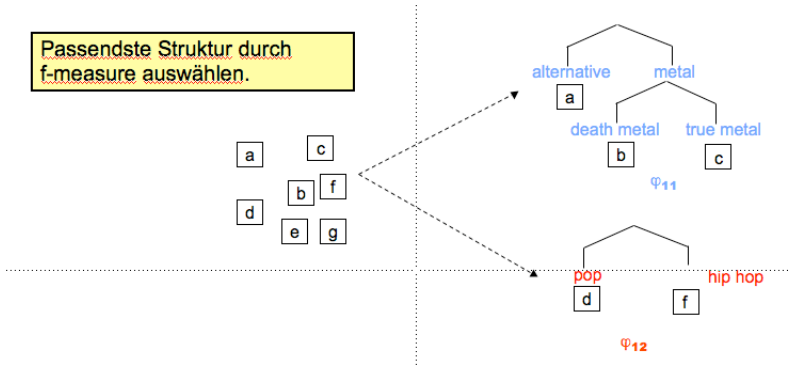
LACE in Bildern - 1: Nicht eingeordnete Stücke, Clusterings anderer Benutzer

Musikstücke durch ID repräsentiert

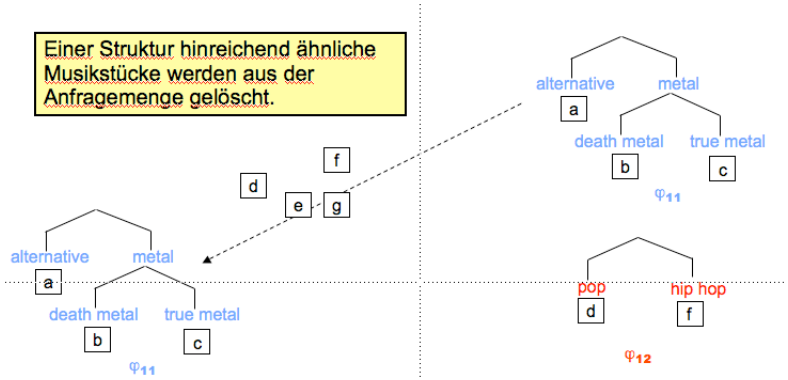




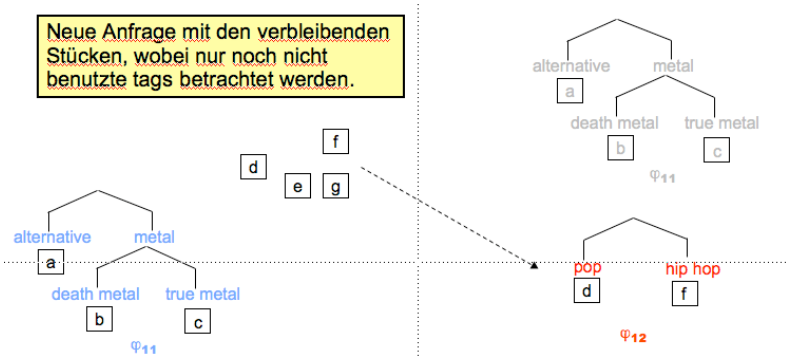
LACE in Bildern - 2: Finden passender Clusterings



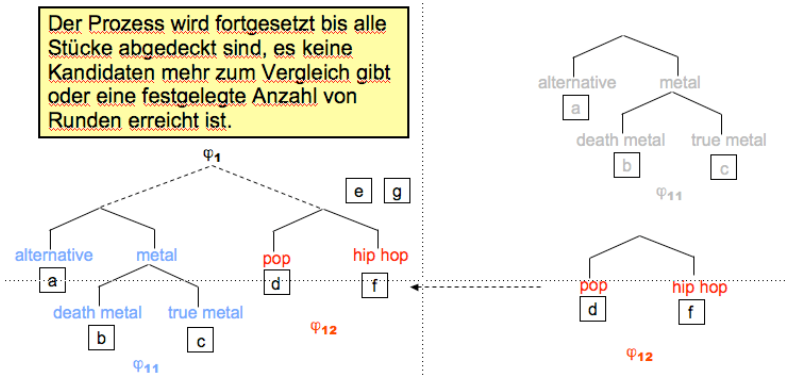
LACE in Bildern - 3: Löschen abgedeckter Stücke



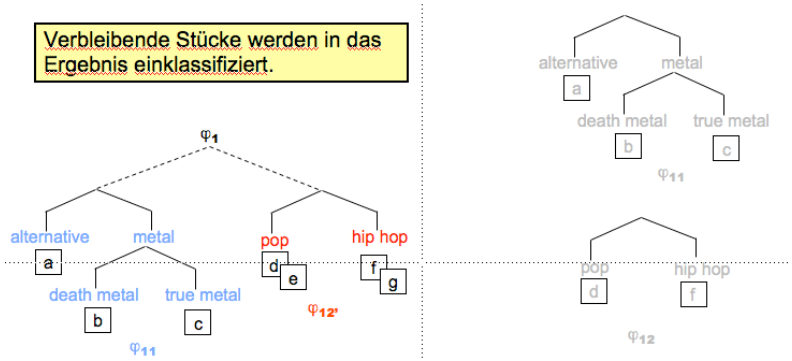
LACE in Bildern - 4: Finden passender Clusterings für den Rest



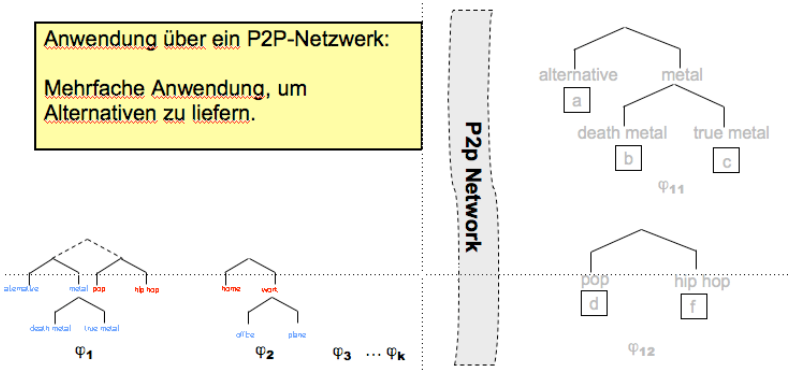
LACE in Bildern - 5: Abbruchbedingung für das sequentielle Abdecken



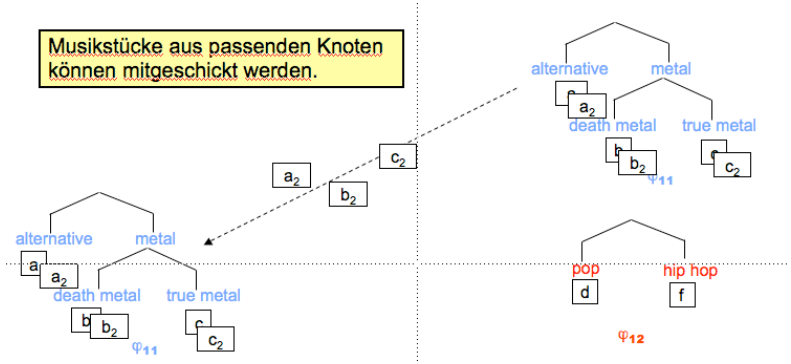
LACE in Bildern - 6: Klassifkation von Stücken in neue Struktur



LACE in Bildern - 7: Posten der abzudeckenden Stücke ins P2P-Netz, Empfangen der passenden Clusterings



Personalisierte Empfehlungen



Qualitätsfunktion für Clustering und Menge von Objekten

- Bei der Repräsentation eines Clusters durch *well-scattered points* ist Z_{φ_i} die Menge von Beobachtungen, die φ_i beschreibt. β sei eine Gewichtung, die Precision und Recall ins Verhältnis setzt:
- Precision:

$$prec(Z_{\varphi_i}, S) = \frac{1}{|Z_{\varphi_i}|} \sum_{\vec{z} \in Z_{\varphi_i}} max\{sim(\vec{x}, \vec{z}) | \vec{x} \in S\}.$$

- Recall:

$$rec(Z_{\varphi_i}, S) = \frac{1}{|S|} \sum_{\vec{x} \in S} max\{sim(\vec{x}, \vec{z}) | \vec{z} \in Z_{\varphi_i}\}.$$

- F-Measure:

$$q_f^*(Z_{\varphi_i}, S) = \frac{(\beta^2 + 1) rec(Z_{\varphi_i}, S) prec(Z_{\varphi_i}, S)}{\beta^2 rec(Z_{\varphi_i}, S) + prec(Z_{\varphi_i}, S)}. \quad (3)$$

Basisalgorithmus Sequenzielles Abdecken

- $O = \emptyset$, $J = I$
- WHILE($|O| < max_{alt}$)
  - $S_u = S$, $B = \emptyset$, $step = 0$
  - WHILE($(S_u \neq \emptyset) \wedge (step < max_{steps})$)
    - $\varphi_i = \arg\max_{\varphi \in J} q^*_f(Z_\varphi, S_u)$
    - $S_u = S_u \setminus \{\vec{x} \in S_u | \vec{x} \sqsubset_\alpha \varphi_i\}$
    - $B = B \cup \{\varphi_i\}$
    - $step = step + 1$
  - $O = O \cup \{bag(B, S)\}$
- Wobei max_{alt} die maximale Anzahl an Alternativen angibt, die Funktion $bag(B, S)$ einen Beutel von Clusterings angibt, der jedem Stück $\vec{x} \in S$ das Clustering $\varphi_i \in B$ zuweist, das die zu $\vec{x}$ ähnlichsten Objekte enthält.

Hierarchisches Vorgehen: Rekursiv Precision und Recall berechnen!

$$prec(Z_{\varphi_i}, S) = \begin{array}{ll} \frac{|Z^*_{\varphi_i}|}{|Z_{\varphi_i}|} prec(Z^*_{\varphi_i}, S) + & updateSchritt \\ \sum_{\varphi_j \prec \varphi_i} \frac{|Z_{\varphi_j}|}{|Z_{\varphi_i}|} prec(Z_{\varphi_j}, S) & direkterNachfolger \end{array}$$

wobei $Z^*_{\varphi_i} = Z_{\varphi_i} \setminus \bigcup_{\varphi_j \prec \varphi_i} Z_{\varphi_j}$ *nur Oberknoten.*

- Die hierarchischen Funktionen φ_j und φ_i, sind in direkter Nachfolgerelation $\varphi_j \prec \varphi_i$, gdw.

$$G_j \subset G_i$$
$$\forall \vec{x} \in S_i : \varphi_j(\vec{x}) = \varphi_i(\vec{x}) \cap G_j$$
$$\neg \exists \varphi'_i : G_j \subset G'_j \subset G_i$$

- Wenn eine optimistische Schätzung des F-measure schon am Wurzelknoten schlechter als ein Schwellwert ist, muss das Clustering nicht weiter untersucht werden!

Daten

- $\varphi_1, ..., \varphi_{39}$ sind 39 Taxonomien für eine Musiksammlung von 1886 Stücken.
- Es wird immer eine Taxonomie weggelassen und auf die restlichen LACE angewandt.
- Das Ergebnis wird mit der weggelassenen Taxonomie verglichen. Differenz der absoluten Tree Distance zwischen zwei Beobachtungen in beiden Taxonomien:

S	x_1	x_2	...	x_m	sum of differences
x_1	-	φ:1;φ':3			2+
x_2		-		φ:1;φ':2	1+
...			-		
x_m				-	
Total					3+

Andere Kriterien und Verfahren

- Andere Kriterien: Korrelation zwischen den Tree Distances FScore:
  - Jedes Cluster der weggelassenen Taxonomie wird mit jedem Cluster der gelernten verglichen (Precision und Recall –> F-measure) und das jeweils beste ausgewählt. Der Durchschnitt ergibt den FScore.
- Single-linkage agglomeratives Clustering
- TD: Rekursives top-down K-Means (Guan, Kulis 2004)
- Mehrfaches Starten, um zu Ensembles zu kommen, von denen stets das beste ausgesucht wird.

Ergebnisse

Method	Correlation	Absolute distance	FScore
LACE	0.44	0.68	0.63
TD ensemble	0.23	2.5	0.55
single-link ensemble	0.17	9.9	0.60
random	0.09	1.8	0.5

Representation	Correlation	Absolute distance	FScore
all points	0.44	0.68	0.63
$\|Z\| = 10$	0.44	0.68	0.63
$\|Z\| = 5$	0.41	0.69	0.63
$\|Z\| = 3$	0.40	0.69	0.62
centroid	0.19	1.1	0.42

Was wissen Sie jetzt?

- Sie haben das Feld der Strukturierung von Sammlungen im Web 2.0 kennen gelernt.
- Sie kennen eine neue Lernaufgabe: lokale alternative Cluster Ensembles und einen Algorithmus dazu.
- Insbesondere haben Sie dabei gesehen, dass man aus der unüberwachten Lernaufgabe des Clusterings manchmal eine halb-überwachte machen kann:
  - Für einzelne Beobachtungen ist angegeben, ob sie im selben oder in verschiedenen Clustern landen sollen (Constraint Clustering).
  - Es soll eine bestimmte Menge von Objekten abgedeckt (strukturiert) werden (LACE).
  - Es soll eine bestimmte Struktur erhalten, aber erweitert werden (Supervised Clustering, LACE).
- Und Sie haben gesehen, wie man Strukturen anderer Benutzer (über ein P2P Netz) nutzen kann.

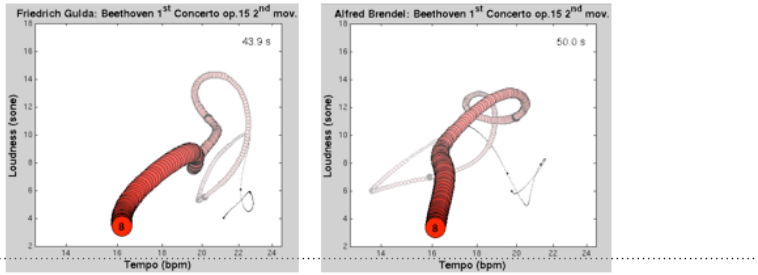
Technische Grundlagen

- Moving Pictures Expert Group Audio Layer 3 Karlheinz Brandenburg, TU Ilmenau, Fraunhofer Institut Standard für Musik und Filme, min. 1/12 komprimiert
- Tauschbörsen für Musik:
  - Napster 80 Mio. Benutzer, Nachfolger: Morpheus, Gnutella, KaZaA
  - KaZaA 500 Mio. Musikstücke
  - Privatsammlungen oft mehr als 10 000 Musikstücke
- Speichern, Abspielen, GUI zum Anbieten von Musik

Arbeitsfelder – Musik

Wissenschaftliche Untersuchung von Musik

– Interpretation (Gerhard Widmer)
Der "Performance Worm": Eine Bewegung des Wurms nach rechts oben beschreibt ein gleichzeitiges Beschleunigen und Lauterwerden. Der dunkelste Punkt repräsentiert den gegenwärtigen Zeitpunkt, die Vergangenheit erscheint blasser. Typische Muster für Künstler finden.



Arbeitsfelder – Music Information Retrieval

- Anfragen: über ID3 tags (Metadaten), query by humming
- Indexierung: über Metadaten, über tags der Benutzer
- Navigation in Sammlungen gemäß Ähnlichkeit
- Klassifikation von Musik
- Empfehlungen

Arbeitsfelder – Intelligente Systeme

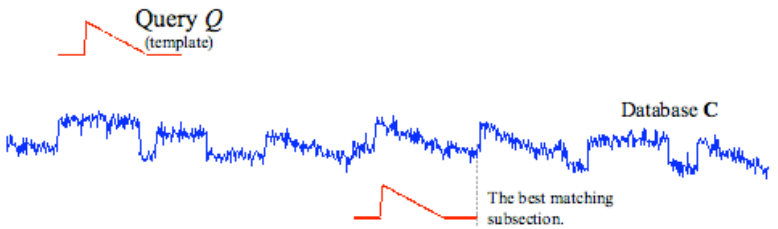
- Automatische Annotation von Musik
- Automatische Klassifikation von Musik nach
  - Genre (nur noch als Benchmark)
  - Benutzerpräferenzen
  - arbiträren tags (Aspekten)
- Automatische Organisation von Sammlungen
- Empfehlungen

Technischer Kern

- Musikdaten sind Zeitreihen der Elongation.
- Wir müssen Ähnlichkeiten von Zeitreihen erkennen. Das ist der technische Kern in fast allen Lernverfahren.
- Ähnlichkeit von Zeitreihen bisher:
  - Ähnlichkeit der Kurven
  - Dynamic Time Warping: Ähnlichkeit mit Verzerrung
- Achtung: Zeitreihenanalyse untersucht eine Zeitreihe und sagt neue Werte in der Zukunft voraus. Hier geht es aber um die Klassifikation oder das Clustering von vielen Zeitreihen. (Eamonn Keough)

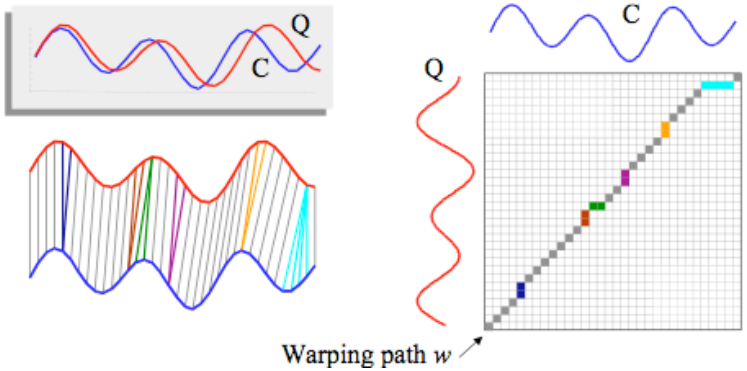
Ähnlichkeit von Zeitreihen

- Gegeben eine Anfrage Q, eine Datenbank mit Zeitreihen C und ein Abstandsmaß,
- finde den Ort in einer Reihe in C, der Q am ähnlichsten ist.



Dynamic Time Warping

$$\gamma(i,j) = d(q_i,c_j) + \min\{ \gamma(i-1,j-1) , \gamma(i-1,j) , \gamma(i,j-1) \}$$



So geht es nicht! Nötig ist die Merkmalsextraktion.

- Musikdaten geben die Ähnlichkeit von Musik nicht wieder. Musik ist nicht ähnlich, wenn die Elongation ähnlich ist.
- Aus den Elongationsdaten müssen Merkmale extrahiert werden, nach denen die Ähnlichkeit bestimmt werden kann.
- Merkmalsextraktion ist die Voraussetzung für:
  - Annotation
  - Indexing
  - Clustering
  - Kassifikation

Merkmalsextraktion

- Eine Reihe von *low level descriptors* wird extrahiert:
  - Lautstärke
  - Peaks, Verhältnis vom höchsten zum zweithöchsten Peak, ...
  - Zero Crossing Rate
  - Spectral Centroid (Cepstral)
  - Mel Frequency Cepstral Coefficient (MFCC)
- Es gibt einen Merkmalssatz, der sich häufig bewährt: Tzanetakis, Dissertation 2002

Ergebnis von Pohle et al. 2005: je Lernaufgabe ist anderer Merkmalssatz nötig!

- Gegeben eine Menge low level descriptors, klassifiziere nach einem Aspekt
  - Genre
  - Stimmung
  - Tempo
  - Instrument vs. Gesang vs. beides
- Es gibt keine Menge von Merkmalen, die alle Klassifikationsaufgaben lösen hilft.
- Je Lernziel (Aspekt) ist ein anderer Merkmalssatz nötig.
- Tzanetakis' Merkmale sind immer einigermaßen gut.

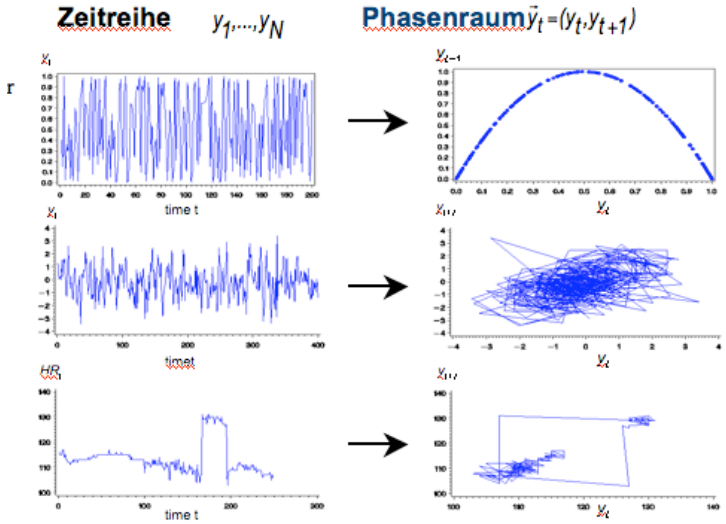
Mierswa Diplomarbeit 2004

- Jeder Mensch achtet auf Unterschiedliches, um Musik zu beurteilen.
- Dieselbe abstrakte Eigenschaft wird anhand völlig unterschiedlicher Merkmale der physikalischen Ebene zugeschrieben.
- Für persönliche Empfehlungen sind auch persönliche Merkmale nötig.
- Also: lernende Merkmalsextraktion für automatische Klassifikation!

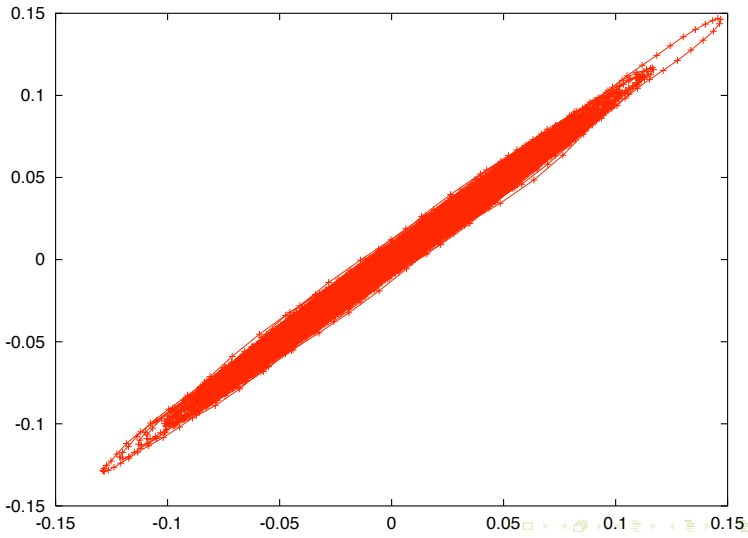
Merkmalsraum strukturieren

- Zeitraum (index)
  - Mittlere Lautstärke: $LS(\vec{x}) = \frac{1}{N}\sum_{i=1} \mid x_i \mid$
  - Tempobestimmung durch Autokorrelation verschobener Reihen: für alle Geschwindigkeiten 90 - 170 bpm: Verschiebung der Reihe um einen Takt, berechnen der Differenz zum Original, wenn die Differenz minimal ist, ist das richtige Tempo bestimmt.
- Frequenzraum
  - Für uns ist die diskrete Fourier-Transformation interessant, insbesondere die schnelle (FFT). Dafür muss die Anzahl der Abtastpunkte eine Zweierpotenz sein. Bei FFT geht die Information verloren, wann die Frequenzen auftreten. Also wird ein Zeitfenster über die Reihe verschoben, innerhalb dessen FFT angewandt wird.
- Phasenraum: gegeben die Messwerte $y_1, ...y_N$ für die Zeitpunkte $1, ..., N$, bilde eine neue Reihe mit den Werten y_{i-1} für die Punkte y_i.

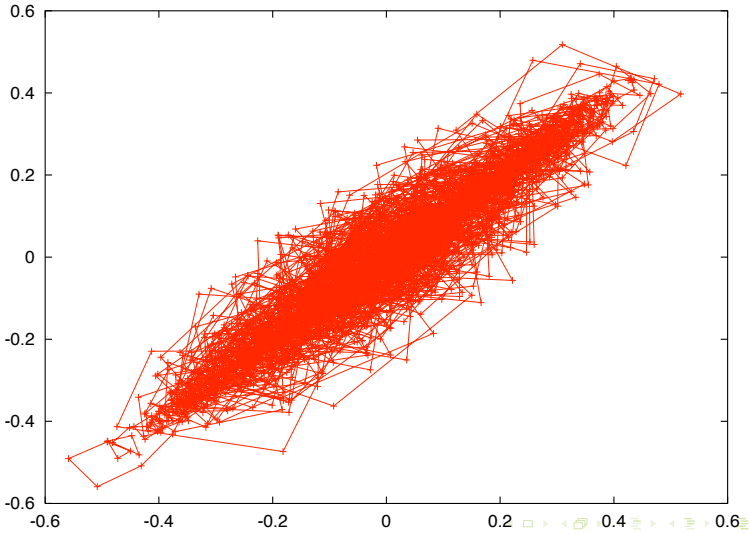
Phasenraum



Phasenraum zur Klassifikation von Genre: Klassik



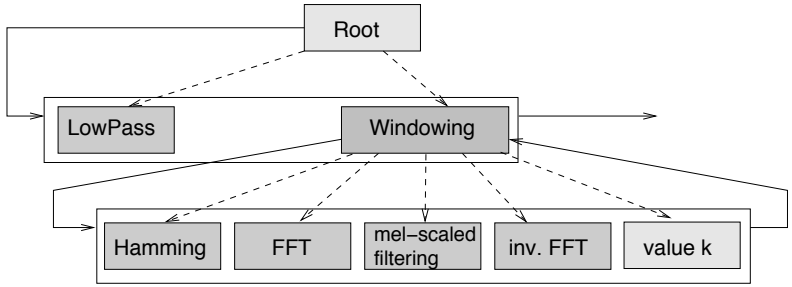
Phasenraum zur Klassifikation von Genre: Pop



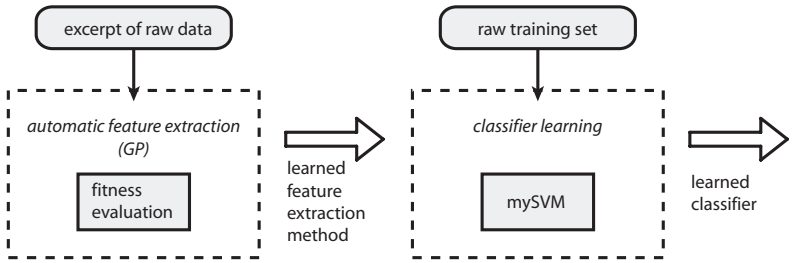
Merkmalsraum weiter strukturieren

- Wir haben die Transformationen im Zeit-, Frequenz-, Phasenraum gesehen.
- Außerdem gibt es Filter und Annotationen von Segmenten.
- Das generalisierte Fenstern trennt die Funktion, die auf Messwerte in einem Fenster angewandt wird, von dem Fenstern selbst. Beim generalisierten Fenstern können beliebig viele beliebige Funktionen auf Werte in einem Fenster angewandt werden.
- Während bei allen vorigen Funktionen wieder eine Reihe zurückgegeben wird, liefert ein Funktional für eine Reihe nur einen Wert zurück.
- Aus diesen modularen Elementen können nun beliebige Merkmalsextraktionen zusammengestellt werden.

Methodenbaum zur Extraktion von MFCC

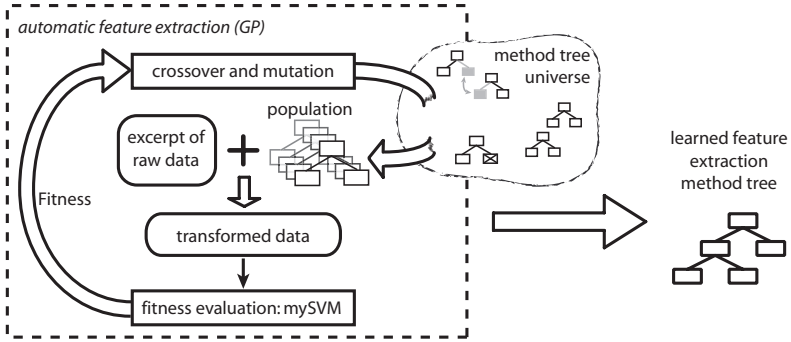


Überblick über den Lernprozess



Mierswa, Morik 2005

Lernen von Methodenbäumen mit genetischer Programmierung



Aufgabenspezifisches Lernen der Merkmale verbessert das Ergebnis

41 Merkmale wurden insgesamt gelernt.

	Classic/pop	**Techno/pop**	**Hiphop/pop**
Accuracy	100%	93.12%	82.50%
Precision	100%	94.80%	85.27%
Recall	100%	93.22%	79.41%
Error	0%	6.88%	17.50%

Tabelle: Klassifikation (lineare SVM) mit gelernten Merkmalen.

	Classic/pop	**Techno/pop**	**Hiphop/pop**
Accuracy	96.50%	64.38%	72.08%
Precision	94.12%	60.38%	70.41%
Recall	95.31%	64.00%	67.65%
Error	3.50%	35.63%	27.92%

Tabelle: Klassifikation mit dem selben Merkmalssatz für alle Aufgaben (lineare SVM).

Klassifikation nach Benutzerpräferenz

- 50 to 80 Stücke Lieblingsmusik
- Die selbe Anzahl negativer Beispiele.

	User$_1$	**User$_2$**	**User$_3$**	**User$_4$**
Accuracy	95.19%	92.14%	90.56%	84.55%
Precision	92.70%	98.33%	90.83%	85.87%
Recall	99.00%	84.67%	93.00%	83.74%
Error	4.81%	7.86%	9.44%	15.45%

Alles implementiert im Value-Series Plugin von RapidMiner.
Verwendbar für alle Wertereihen!

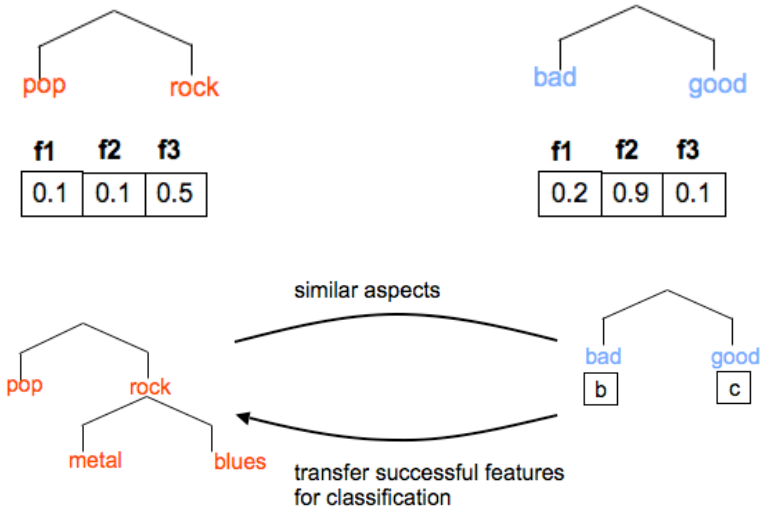
Eigenschaften lernender Merkmalsextraktion

- Sehr gute Lernergebnisse
- Aufwand des Benutzers, die Beispielmengen zusammenzustellen → automatisch (aus Hörverhalten) extrahieren!
- Aufwand der genetischen Programmierung
- Merkmale werden aus einem Musikstück (Sample) extrahiert – funktioniert nicht inkrementell (online).

Merkmalsübertragung

- Wenn das Trainieren der Merkmalsextraktion so lange dauert (1 Woche), sollte für ähnliche Lernaufgaben auch bereits gelernte Merkmalssätze verwendet werden (Mierswa/Wurst 2005, Wurst/Morik 2006).
- Charakterisierung einer Merkmalsmenge durch Gewichtung von Basismerkmalen.
- Feststellen der Eignung von Basismerkmalen für eine Klassifikationsaufgabe.
- Ähnliche Gewichte der Basismerkmale → ähnliche Lernaufgaben und Transfer des gesamten Merkmalssatzes.

Merkmalstransfer im Bild



Eignung von Merkmalen für eine Lernaufgabe

- Ein Merkmal X_{ik} ist **irrelevant** für eine Klassifikationsaufgabe t_i, wenn es nicht mit Y_i korreliert ist: $Pr(Y_i \mid X_{ik}) = Pr(Y_i)$.
  Die Menge irrelevanter Merkmale für t_i ist IF_i.
- Zwei Merkmale X_{ik} und X_{ir} heißen **alternativ** bzgl. einer Lernaufgabe t_i, $X_{ik} \sim X_{ir}$, gdw. $X_{ir} = a + b \cdot X_{ik}, b > 0$.
  Die Menge alternativer Merkmale für t_i ist AF_i.
- X_B sei eine Menge von Basismerkmalen.
- Die Merkmale sollen nun so gewichtet werden, wie es ihrer Eignung für die Lösung einer Lernaufgabe enspricht $w : X_B \to \mathcal{R}$.

Bedingungen für Merkmalsgewichtungen, die die Charakterisierung von Lernaufgaben erlauben

1. $w(X_{ik}) = 0$, wenn $X_{ik} \in X_B$ irrelevant ist.
   Irrelevante Merkmale sind mit 0 gewichtet.
2. Für $AF_i \subseteq X_B$ gilt:
   $\forall S \subset AF_i, S \neq \{\} : \sum_{X_i \in S} w(X_k) = \sum_{X_k \in AF_i} w(X_k) = \hat{w}$
   Die Gewichtsumme alternativer Merkmale ist unabhängig von der Anzahl alternativer Merkmale.
3. $X_{ik} \sim X_{ir} \Rightarrow w(X_{ik}) = w(X_{ir})$
   Alternative Merkmale sind gleich gewichtet.
4. $\forall X_{ik} \in AF_i : X_{ir} \in IF_i \vee \exists X_{ir} \in X_B : X_{ik} X_{ik} \sim X_{ir} \Rightarrow$
   $\forall X_{ir} \in X_B : \nexists X_{ik} \in AF_i : X_{ir} \sim X_{ik} \wedge w'(X_{ir}) = w(X_{ik})$
   mit $w' : X_B \cup AF \to \mathcal{R}$.
   Eine Menge alternativer Merkmale ist nicht stärker gewichtet als ein einzelnes Merkmal.

Die Bedingungen gelten nicht immer!

- Alle Methoden der Merkmalsauswahl, die Merkmale binär gewichten, verletzen Bedingung 2 oder 3, sobald ein alternatives Merkmal hinzugefügt wird.
  $X'_B = X_B \cup \{X_{ir}\}, X_{ir} \sim X_{ik}, X_{ik} \in X_B \Rightarrow$
  $w'(X_{ir}) = w'(X_{ik}) = w(X_{il}) = 1$ weil ein ausgewähltes Merkmal in X_B Gewicht 1 hat; Verletzung 2. Bedingung: die Summe wäre 2!
  oder $w'(X_{ir}) \neq w(X_{ik})$ Verletzung 3. Bedingung (Alternativen sind gleichgewichtet).
- Jede Methode, die die Merkmale unabhängig voneinander gewichtet, verletzt Bedingung 2. Bei $X'_B = X_B \cup \{X_{ir}\}$ bleiben alle Gewichte für Merkmale in X_B gleich. Wenn $X_{ir} \sim X_{ik}, X_{ik} \in X_B$ verändert sich die Summe, so dass 2. Bedingung verletzt ist.

Die lineare SVM erfüllt alle Bedingungen

Die Merkmalsgewichtung durch die lineare SVM, $\vec{\beta}$, erfüllt alle Bedingungen.

- Bedingung 1: Die Euklidsche Länge von $\vec{\beta}$ soll minimiert werden, also werden möglichst Merkmale mit 0 gewichtet, wenn dadurch nicht der Fehler steigt. Also werden irrelevante Merkmale mit 0 gewichtet.
- Bedingung 2: Fügen wir einfach das selbe Merkmal mehrfach hinzu, so ergibt sich $(\beta_{i1} + ... + \beta_{im})\vec{x}$ in $\vec{\beta}\vec{x} + \beta_0$. Die optimale Hyperebene ändert sich nicht und die Summe der Gewichte bei allen anderen Merkmalen bleibt unverändert.
- Bedingung 3: Die Summe der alternativen Merkmale verteilt sich gleichmäßig auf die Alternativen.
- Bedingung 4: Folglich ist die Menge alternativer Merkmale nicht stärker gewichtet als ein einzelnes Merkmal.

Geeignete Abstandsmaße für die Gewichtung der Basismerkmale als Ähnlichkeit von Lernaufgaben

Das Abstandsmaß $d : T \times T \rightarrow \mathcal{R}^+$ soll erfüllen:

1. $d(\vec{t_1}, \vec{t_2}) = 0 \Leftrightarrow \vec{t_1} = \vec{t_2}$
2. $d(\vec{t_1}, \vec{t_2}) = d(\vec{t_2}, \vec{t_1})$
3. $d(\vec{t_1}, \vec{t_2}) = d(\vec{t_1}', \vec{t_2}')$, $\vec{t_1}' = \vec{t_1}, \vec{t_1}' \in X_B^2 \cup IF_1^2$ und $\vec{t_2}' = \vec{t_2}, \vec{t_2}' \in X_B^2 \cup IF_2^2$
   gleiche Gewichtsvektoren behalten im erweiterten Bereich gleichen Abstand.
4. $d(\vec{t_1}, \vec{t_2}) = d(\vec{t_1}', \vec{t_2}')$, $\vec{t_1}' = \vec{t_1}, \vec{t_1}' \in X_B^2 \cup AF_1^2$ und $\vec{t_2}' = \vec{t_2}, \vec{t_2}' \in X_B^2 \cup AF_2^2$
   gleiche Gewichtsvektoren behalten im erweiterten Bereich gleichen Abstand.

Die Bedingungen gelten nicht immer!

Bei Euklidschem Abstand wird Bedingung 5 nicht eingehalten, d.h. das Hinzufügen alternativer Merkmale verändert den Abstand.

- Das alternative Merkmal X_r wird X_B hinzugefügt und ist alternativ zu $X_k \in X_B$. Wenn die Bedingungen an die Merkmalsgewichtung eingehalten sind, gilt:
  $w'(X_{sk}) = w'(X_{sr}) = \frac{w(X_{sk})}{2} = \frac{w(X_{sr})}{2}$ für $s = 1, 2$
- Seien alle anderen Merkmalsabstände S, dann ist

$$\begin{aligned} d(\vec{t_1}', \vec{t_2}') &= \sqrt{S + 2(w'(X_{ik}) - w'(X_{jk}))^2} \\ &= \sqrt{S + 2(\tfrac{w(X_{ik})}{2} - \tfrac{w(X_{jk})}{2})^2} \\ &= \sqrt{S + \tfrac{1}{2}(w(X_{ik}) - w(X_{jk}))^2} \\ &\neq \sqrt{S + (w(X_{ik}) - w(X_{jk}))^2} \\ &= d(\vec{t_1}, \vec{t_2}) \end{aligned}$$

Manhattan Abstand hält alle Bedingungen ein

- Bedingungen 1 - 3 sind die einer Metrik.
- Bedingung 4: Wir fügen ein für beide Lernaufgaben $\vec{t_1}, \vec{t_2}$ irrelevantes Merkmal X_{k+1} hinzu. Wenn die Bedingung 4 an die Gewichtung eingehalten ist, gilt: $\mid w'(X_{1,k+1}) - w'(X_{2,k+1}) \mid = 0$. Also:

$$\begin{aligned} d(\vec{t_1}', \vec{t_2}') &= \sum_{r=1}^{k} \mid w'(X_{1,r}) - w'(X_{2,r}) \mid + 0 \\ &= d(\vec{t_1}, \vec{t_2}) \end{aligned}$$

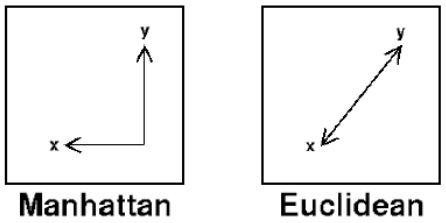
Manhattan Fortsetzung

- Bedingung 5: Das alternative Merkmal X_{k+1} wird X_B hinzugefügt und ist alternativ zu $X_k \in X_B$. Wenn die Bedingungen an die Merkmalsgewichtung eingehalten sind, gilt: $w'(X_{s,k+1}) = w'(X_{s,k}) = \frac{w(X_{s,k+1})}{2} = \frac{w(X_{s,k})}{2}$ für $s = 1, 2$

$$\begin{aligned} d(\vec{t_1}', \vec{t_2}') &= (\sum_{r=1}^{k-1} \mid w'(X_{1,r}) - w'(X_{2,r}) \mid) + \\ &\quad 2(\mid w'(X_{1,k+1}) - w'(X_{2,k+1}) \mid) \\ &= (\sum_{r=1}^{k-1} \mid w(X_{1,r}) - w(X_{2,r}) \mid) + \\ &\quad \mid w(X_{1,k}) - w(X_{2,k}) \mid \\ &= d(\vec{t_1}, \vec{t_2}) \end{aligned}$$

Unterschied der Abstandsmaße Manhattan und Euklid

$d(x, y)$



Anwendung der Merkmalsübertragung

- Gegeben die 39 Taxonomien zur Musikorganisation. Je Knoten sei die Lernaufgabe, in die Unterknoten zu klassifizieren.
- Wir optimieren Musikmerkmale für jede Lernaufgabe.
- Als Basismerkmale werden 10 gewählt, die für die meisten Lernaufgaben erzeugt wurden.
- Anwendung der linearen SVM auf jede Lernaufgabe liefert $\vec{\beta}$ und damit auch eine Gewichtung der Basismerkmale. $O(\mid X_B \mid\mid T \mid N^3)$
- Gemäß der gewichteten Basismerkmale wird die Ähnlichkeit der Lernaufgaben festgestellt. $O(\mid X_B \mid\mid T \mid^2)$
- Bei ähnlichen Lernaufgaben wird der komplette Merkmalssatz transferiert.

Ergebnis des Merkmalsübertragung

	Accuracy	Time	Optimization cycles
base features	0.79	-	-
optimal features	0.92	42s	3970
cbfc (k = 1)	0.85	3s	257
cbfc (k = 3)	0.88	5s	389
cbfc (k = 9)	0.89	8s	678

Tabelle: Durchschnittliche accuracy und Gesamtaufwand auf einem Testset von 11 Taxonomien für Lernen mit Basismerkmalen, optimierten Merkmalssätzen und Merkmalstransfer von den k ähnlichsten Lernaufgaben (cbfc).

Was wissen Sie jetzt?

- Merkmale können aus Basisfunktionen und -transformationen per Genetischer Programmierung hergestellt werden, wobei die Qualität des Lernergebnisses optimiert wird.
- Merkmale werden von Entscheidungsbaumlernern und der SVM sehr unerschiedlich behandelt. Wichtigster Unterschied ist die Behandlung irrelevanter oder alternativer Merkmale.
- Nur die SVM-Merkmalsgewichtung im Zusammenhang mit der Manhattan-Distanz ermöglicht, anhand der Gewichtung von Basismerkmalen die Ähnlichkeit von Lernaufgaben festzustellen.
- Antrainierte Merkmalssätze können auf ähnliche Lernaufgaben übertragen werden und liefern dann mit viel weniger Aufwand fast gleich gute Ergebnisse.