

**Knowledge Discovery in Databases II**  
 SoSe 2009

**Übungsblatt 6: Ensemble-Techniken, Multi-Repräsentiertes Data Mining**

Besprechung am Donnerstag, 18.06.2009

**Aufgabe 6-1** *Error Correcting Output Codes*

- (a) Beschreiben Sie das Klassifikationsschema *one-versus-rest* für ein 4-Klassen-Problem in der Notation, die Sie für ECOCs kennengelernt haben.
- (b) Beschreiben Sie ein ECOC-Schema für eine minimale Anzahl von Base-Classifiern an sowie ein vollständiges ECOC-Schema, das die Codes für jede nicht-triviale Aufteilung der 4 Klassen in ein zwei-elementige Menge von Klassen angibt.  
 Was beobachten Sie für die Row-Separation?

**Aufgabe 6-2** *Ensemble Multi-Klassen-Klassifikation*

In der Vorlesung haben Sie die Ensemble-Techniken *one-versus-rest*, *all-pairs* und *ECOC* kennengelernt, die Klassifikationsprobleme mit mehr als 2 Klassen auf mehrere 2-Klassen-Probleme zurückführen. Für *one-versus-rest* und *all-pairs* können wir in der Test-/Anwendungsphase ein einfaches Mehrheitsvoting für die Klassifikationsentscheidung annehmen. Für *ECOC* haben wir die Entscheidungsregel genauer diskutiert. Eine weitere Möglichkeit stellt das DDAG-Schema dar: Aus den einzelnen *all-pairs*-Klassifikatoren wird ein gerichteter, azyklischer Graph für die Klassifikationsentscheidung gebildet (DDAG=Decision Directed Acyclic Graph), siehe Abbildung 1.

- (a) Welche Vor- oder Nachteile hat diese Strategie gegenüber dem Voting über alle paarweisen Klassifikatoren?

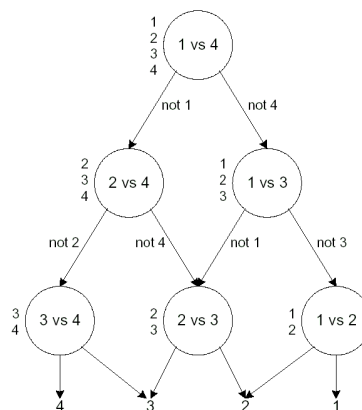


Abbildung 1: Klassifikationsschema DDAG

- (b) Nehmen Sie als Komplexität eines Base-Classifiers im Training die Funktion  $t : \mathbb{N} \rightarrow \mathbb{R}_0^+$  an, die abhängig von der Anzahl der Trainingsbeispiele ist. Wie verhalten sich die unterschiedlichen Schemata hinsichtlich ihres Zeitbedarfs in der Trainingsphase bei  $n$  Klassen und  $m$  Beispielen für jede Klasse? Wie sieht es in der Anwendungsphase aus, wenn Sie einen konstanten Zeitbedarf für die Vorhersage des einzelnen Base-Klassifikators annehmen?

### Aufgabe 6-3 *Kombination von zwei Ähnlichkeitsmaßen*

Gegeben seien zwei Kernels  $k_1$  und  $k_2$ . Wir kombinieren sie in einen gemeinsamen Kernel  $k_{com}$

$$k_{com} = \alpha k_1 + (1 - \alpha) k_2, \quad (1)$$

wobei  $\alpha \in [0; 1]$ .

Wir wenden den Kernel  $k_{com}$  auf zwei Klassifikationsprobleme an, wobei wir das Experiment für verschiedene Werte von  $\alpha$  wiederholen. In der folgenden Abbildung stellen wir die Klassifikationsgenauigkeit auf dem ersten Datensatz (R1) und auf dem zweiten Datensatz (R2) in Abhängigkeit von  $\alpha$  dar:

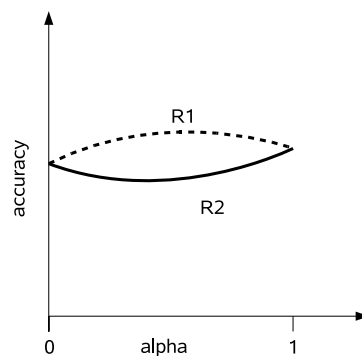


Abbildung 2: Classification accuracy vs.  $\alpha$

Beantworten Sie folgende Fragen anhand von Abbildung 2:

- Auf welchem der beiden Datensätze lohnt sich die Kombination der beiden Kernels?
- Wann funktionieren die Kernels  $k_1$  und  $k_2$  alleine besser als kombiniert?

### Aufgabe 6-4 *Multi-Repräsentierte Klassifikation*

Gegeben sei ein Datensatz mit multiplen Repräsentationen jedes Datenobjekts. Wir möchten Klassenzugehörigkeiten mittels dieser multiplen Repräsentationen ermitteln.

- In welcher Phase des Klassifikationsprozesses können wir die verschiedenen Repräsentationen integrieren?
- Wie können wir die multiplen Repräsentationen beim Trainieren integrieren?
- Wie können wir die multiplen Repräsentationen beim Vorhersagen integrieren?
- Ist in beiden Fällen zuvor eine Normalisierung erforderlich?