JProf. Dr. Uwe Ligges, Prof. Dr. Katharina Morik
Julia Schiffner, Felix Jungermann

Dortmund, 11. Mai
Abgabe: bis zum 18.05. um 10:00 Uhr an `felix.jungermann@tu-dortmund.de`

Übungen zur Vorlesung

Wissensentdeckung in Datenbanken

Sommersemester 2010

Blatt 4

Für die Lösung dieses Übungszettels benötigen Sie wieder das mittlerweile bekannte Programm RapidMiner. Falls Sie Hilfe zu RapidMiner benötigen oder sonstige Fragen zum Übungszettel haben, können Sie sich gerne per E-Mail an mich wenden. Schauen Sie aber zuerst in die Dokumentation von RapidMiner — auf der Webseite des Herstellers gibt es ein Tutorial und ein GUI-Tutorial! Außerdem finden sich viele Beispiele im Samples-Repository. Vergessen Sie auch nicht, in RapidMiner den Expertenmodus einzustellen (Person mit bzw. ohne Hut in der Symbolleiste).

Aufgabe 4.1 (5 Punkte)

In dieser Aufgabe sollen Sie wieder den schon bekannten Iris-Datensatz untersuchen. Der Datensatz enthält bekannterweise jeweils 50 Beispiele zu drei verschiedenen Typen von Iris-Pflanzen (Iris Setosa, Iris Versicolour und Iris Virginica). Die vier Attribute beschreiben Länge und Breite von Kelch- und Blütenblättern.

(a) Lesen Sie die Daten aus dem `Samples`-Repository ein. Erstellen Sie in der Plot-Ansicht von RapidMiner eine Scatter-Matrix (Plots = `class`) und speichern Sie die Matrix als PNG-Grafik (gehört mit zur Abgabe). Beurteilen Sie die Daten bezüglich ihrer linearen Separierbarkeit. Welche Kombination von Attributen eignet sich am schlechtesten/besten, um die Daten linear voneinander zu trennen? Lassen sich die Daten linear voneinander trennen?

(b) Wir sind nun an der Performanz von k-NN für $k = 1$ interessiert. Fügen Sie nach dem Einlesen der Daten als Lerner zunächst den Operator `k-NN` in Ihr Experiment ein ($k = 1$), wenden Sie das Modell auf den Datensatz an (`Apply Model`) und bestimmen Sie die Performanz mit dem Operator `Performance (Classification)`. Welche Werte erhalten Sie für die `accuracy`, nachdem Sie das Experiment gestartet haben? Deuten Sie das Ergebnis! (Um Datensätze mehrmals zu benutzen, müssen Sie den `Multiply`-Operator anwenden!)

(c) Bestimmen Sie nun die Accuracy für unterschiedliche k über eine 5-fache Kreuzvalidierung. Betten Sie hierfür die Operatoren (bis auf `Retrieve` und `Mulitply`) der vorangegangenen Aufgabe in den `X-Validation`-Operator ein. Befassen Sie sich zudem

mit dem Operator `Loop Parameters`, um die Accuracy für verschiedene Werte von k zu bestimmen. Machen Sie die `X-Validation` zu einem Kind des vorgenannten Operators. Wählen Sie über `Edit Parameter Settings ...` den Parameter `k` von `k-NN` als veränderlichen Parameter aus mit Min = 1, Max = 79, Steps = 39 und Scale = linear.

Um die Accuracy für unterschiedliche Werte von k loggen zu können, fügen Sie außerdem als letztes Kind von `Loop Parameters` den Operator `Log` ein. Über `Edit List ...` und den `Add`-Knopf können Sie nun auf der linken Seite (`log`) Namen für die zu loggenden Werte vergeben (z. B. `k` und `accuracy`) und auf der rechten Seite (`column_name`) abhängig vom Operator den zugeordneten veränderlichen Parameter bzw. Wert auswählen.

Wenn das Experiment durchgelaufen ist, erhalten Sie als Ergebnis von `ProcessLog` eine Tabelle. Wechseln Sie auf die Plot-Ansicht und erstellen Sie einen Scatter-Plot für die Werte von `k` und `accuracy`. Speichern Sie den Plot (gehört mit zur Abgabe) und interpretieren Sie das Ergebnis!

Aufgabe 4.2 (5 Punkte)
Das Verfahren k-NN klassifiziert Beispiele anhand der Klassen von Datenpunkten in einer Nachbarschaft. Die k nächsten Nachbarn eines Beispiels werden dabei oft anhand ihres euklidischen Abstands zum Beispiel bestimmt. In dieser Aufgabe soll experimentell untersucht werden, wie die Anzahl der Attribute die Abstände zwischen Datenpunkten beeinflussen kann.

RapidMiner kann Daten nicht nur einlesen, sondern auch generieren. Erstellen Sie ein Experiment mit den aufeinander folgenden Operatoren `Generate Data` und `Data to Similarity`. Wählen Sie bei allen Teilaufgaben für den Parameter `target_function` den Wert `random` aus (Gleichverteilung) und lassen sie jeweils 250 Beispiele mit Attributwerten im Intervall $[0, 1]$ generieren.

(a) Sei p die Anzahl der Attribute. Berechnen Sie (per Hand) für p = 2, 5, 10, 20, 50, 75, 100, 150, 200, 250 den maximal möglichen euklidischen Abstand $d_{\max}$ zwischen den Datenpunkten. Die Attributwerte können dabei aus dem oben angegebenen Bereich stammen. Tip: Erstellen Sie für die unterschiedlichen Werte eine Tabelle, in die sie auch die Werte der nachfolgenden Teilaufgaben eintragen können. Berechnen Sie alle Werte auf zwei Nachkommastellen genau.

(b) Starten Sie das Experiment jeweils für die zuvor genannten Werte von p und lassen Sie sich das Histogramm der vom Operator `Data to Similarity` paarweise berechneten Distanzen ausgeben (Karteireiter `SimilarityMeasureObject`). Diese Plots müssen nicht mit abgegeben werden. Notieren Sie für jedes p nach Augenmaß die kleinste (d_1) und größte (d_2) Distanz mit einer Häufigkeit > 0 und berechnen Sie $d = d_2 - d_1$.

(c) Sie sollten nun eine Tabelle vorliegen haben, die für jedes p die Werte $d_{\max}$ und d enthält. Der Wert von d gibt dabei die Größe/Spanne des Bereichs an, in dem die Distanzen für ein bestimmtes p tatsächlich liegen. Berechnen Sie für jedes p den Anteil von d an $d_{\max}$ in Prozent. Was fällt Ihnen auf? Beurteilen Sie Ihre Ergebnisse in Hinblick auf die Fähigkeit von k-NN, hochdimensionale Daten korrekt zu klassifizieren.

Geben Sie bitte Ihre RapidMiner-Experiment-Dateien mit ab!