Alexander Hinneburg,
hinneburg@informatik.uni-halle.de

23.11.2005

Übung zur Vorlesung „Data Mining in Datenbanken“

Übungsblatt 4[1]

Data Cleaning und Vorbereitungen
Abgabe am 17.11.2005

Die letzte Übung beschäftigte sich mit einer ausgewählten Teilmenge der 50 besten Gene einer Leukämie-Datenmenge. In dieser Übung sollen Sie wie ein richtiger Data Miner vorgehen und mit den tatsächlichen Ausgangsdaten arbeiten. Sie werden lernen, dass der Data Mining Prozeß oft aus vielen kleinen Schritten besteht, die alle korrekt durchgeführt werden müssen, um brauchbare Ergebnisse zu erzielen.
Laden Sie die Daten `ALL_AML_original_data.zip` von der Vorlesungsseite und entpacken Sie das Archiv. Sie bekommen

- Traingingsdaten: `data_set_ALL_AML_train.txt`
- Testdaten: `data_set_ALL_AML_independent.txt`
- Klassendaten: `table_ALL_AML_samples.txt`

Benennen Sie die Trainingsdatei um in `ALL_AML_grow.train.orig.txt` und die Testdatei in `ALL_AML_grow.test.orig.txt` .
Konvention: nutzen Sie das gleiche Präfix für Dateien ähnlichen Typs und markieren Sie die verschiedenen Versionen im hinteren Teil des Namens. Das Kürzel *orig* steht für Originaldatei und *grow* für Gene in Zeilen (row). Die Extension .tmp steht für temporäre Dateien.
Einen Artikel (Todd Golub et. al: Molecular Classification of Cancer: Class Discovery and Class Prediction by Gene Expression Monitoring) mit diesen Daten finden Sie über die Vorlesungsseite.
Die Trainings- (78 Felder) und Testdaten (70 Felder) haben beide 7130 Zeilen. Die ersten beiden Felder sind `Gene Description` (eine lange Beschreibung wie z.B. `GB DEF = PDGFRalpha protein`) und `Gene Accession Number` (ein kurzer Name wie `X95095_at`). Die restlichen Felder sind Paare bestehend aus Patientennummer (z.B. 1,2,..38) und einem Affymetix „call“ (`P` Gen ist präsent, `A` falls absent, `M` falls marginal).
Stellen Sie sich die Trainingsdaten wie eine Tabelle mit 7130 Zeilen und 78 Spalten vor. Dies ist das übliche Format für Microarraydaten, für Data Mining brauchen Sie die transponierte Tabelle (Zeilen = Patienten, Gene=Spalten).

Aufgabe 1 Führen Sie die folgenden Schritte zum Reinigen der Daten auf den Test- und Trainingsdaten durch. Nutzen Sie dafür Unix Tools, Skripte oder andere Programme. Nutzen Sie möglichst Standardprogramme und vermeiden Sie so weit wie möglich eigene Programme zu schreiben (z.B. mit `wc` können Sie die Anzahl der Zeilen feststellen). Unter

[1] Achten Sie auch auf die Form Ihrer Lösungen, z.B. daß jedes Bild eine Bildunterschrift hat. Bitte geben Sie keine losen Blätter ab. Um die Übungen in angemessener Zeit zu diskutieren, bereiten Sie neben Ihrer Lösung ein kleine Präsentation vor, mit der Sie dann Ihre Ergebnisse zeigen können.

Unix oder Cygwin (Windows) wird die Hilfe zu einem Kommando mit `man Kommando` abgefragt und wenn Sie das Kommando nicht wissen mit `man -k Stichwort`. Nützliche Kommandos sind `wc`, head, tail, sort, cut, join, tr, grep, paste, cut, awk, sed. Wichtig sind auch Aus- (`>`)und Einlenkung (`<`), bzw. Weiterleitung (`|`).

Dokumentieren Sie alle Schritte und erzeugen Sie Dateien für Zwischenergebnisse. Ein Kommilitone sollte ihre Dokumentation und den Sinn dahinter verstehen können. Geben Sie nach jedem Schritt die Anzahl der Zeilen und Spalten der Dateien an.

1. Löschen Sie die initialen Einträge mit Genenbeschreibungen, die „control“ enthalten. (Das sind Affymetrix Kontrollen, keine menschlichen Gene). Nennen Sie diese Dateien `ALL_AML_grow.train.noaffy.tmp` und `ALL_AML_grow.test.noaffy.tmp` . Hinweis: Sie können das Unix Kommando `grep` benutzen um diese Einträge zu finden. Protokollieren Sie, wie viele Kontrolleinträge Sie gefunden haben.
2. Löschen Sie das erste Feld (lange Beschreibung) und die „call“ Felder (es bleiben nur die Felder 1,2,3,5,7,9,...). Hinweis: nutzen Sie das Unix Kommando `cut` um dies zu tun.
3. Ersetzen Sie alle tabs durch Kommata (dieser Schritt muß später vielleicht wiederholt werden).
4. Ändern Sie `Gene Accession Number` in `ID` im ersten Eintrag. (Bemerkung: Dadurch sollen mögliche Probleme vermieden werden, die manche DM Tools mit Leerzeichen in Bezeichnern haben.)
5. Normalisieren Sie die Daten: für alle Attribute soll das Minimum 20 und das Maximum 16000 sein ($x < 20 \Rightarrow x = 20$, $x > 16000 \Rightarrow x = 16000$). (Expressionswerte unter 20 und über 16000 wurden von den Biologen als unzuverlässig bei diesem Experiment eingeschätzt). Nennen Sie die erzeugten Dateien `ALL_AML_grow.train.norm.tmp` und `ALL_AML_grow.test.norm.tmp` .
6. Schreiben Sie ein kleines Java oder Shell Skript zum Transponieren der Tabelle. Nennen Sie die Dateien `ALL_AML_gcol.test.tmp` und `ALL_AML_gcol.train.tmp` (`gcol` steht für Gene in Spalten). Diese Dateien sollten 7071 Spalten und 39 Zeilen in `train`, 35 Zeilen in `test` haben.
7. Extrahieren Sie aus der Datei `table_ALL_AML_samples.txt` die Dateien `ALL_AML_idclass.train.txt` und `ALL_AML_idclass.test.txt` mit den Spalten `ID` und `class`. Hier eignet sich eine Kombination aus Unix Kommandos und, wenn Sie es nicht anders hinbekommen, manuellem Editieren. Fügen Sie eine Kopfzeile mit `ID,Class` zu jeder Datei.

   Danach sollte die Datei `ALL_AML_idclass.train.txt` 39 Zeilen und zwei Spalten haben. Die erste Zeile (Kopf) ist `ID,Class`, die nächsten 27 Zeilen haben Klasse `ALL` und die letzten 11 Zeilen haben Klasse `AML`. `ALL_AML_idclass.test.txt` sollte 20 `ALL` Sample und 14 `AML` Samples gemischt haben.
8. Beachten Sie, daß die Sample IDs in den `ALL_AML_gcol*.txt` Dateien in anderer Reihenfolge als in den `*idclass` Dateien sind. Nutzen Sie, Unix Kommandos um eine kombinierte Datei `ALL_AML_gcol_class.train.csv` und `ALL_AML_gcol_class.test.csv` zu erzeugen, wo `ID` als erstes Feld und `class` als letztes Feld erscheinen; die Gen-Expressionsfelder sind dazwischen. Stellen Sie sicher, daß alle überflüssigen Leerzeichen und Tabs entfernt wurden und die Werte

durch Kommata getrennt sind. Nennen Sie der fertigen Dateien von `.txt` in `.csv` um.

Sie bekommen Zusatzpunkte, wenn Sie ohne manuelles Editieren auskommen und alle Schritte in einem dokumentierten ausführbaren Skript zusammenfassen.

Aufgabe 2 Wie in Übung 2, konvertieren Sie `ALL_AML_gcol_class.train.csv` in `ALL_AML_allgenes.train.arff` und `ALL_AML_gcol_class.test.csv` in `ALL_AML_allgenes.test.arff`.

1. Benutzen `ALL_AML_allgenes.train.arff` als Trainingsdaten und `ALL_AML_allgenes.test.arff` als Testdaten, lernen Sie den OneR Klassifikator. Welche Genauigkeit erhalten Sie?
2. Entfernen Sie dann das Feld `ID`, und lernen Sie erneut OneR, NaiveBayes Simple und J48, mit der Option `using training set only`. Was sind die Fehlerraten für die verschiedenen Methoden, interpretieren und vergleichen Sie die gelernten Klassifikatoren? Warnung: einige Methoden könnten sehr lange laufen oder Fehler ausgeben, wegen der extrem vielen Attribute.
3. Wenn Sie bis hierhin gekommen sind, **Herzlichen Glückwunsch!** Aufgrund ihrer gemachten Erfahrungen, welche drei Dinge sind wichtig im Data Mining Prozess? Warum?