Maschinelles Lernen: Symbolische Ansätze

TECHNISCHE UNIVERSITÄT DARMSTADT

Wintersemester 2010/2011
Projekt: Abgabe am 19.02.2011

Ziel des Projektes ist es, praktische Erfahrungen im Maschinellem Lernen zu sammeln. Hierzu sollen mehrere Projektaufgaben mit Hilfe des Machine Learning Frameworks Weka gelöst werden. Das Projekt kann allein bzw. in einer Kleingruppe (maximal 3 Studenten, Name + Matrikelnr. müssen in der Abgabe ersichtlich sein) bearbeitet werden. Die Abgabe soll in einem üblichen Dokumentenformat (z.B. Beamer, PowerPoint o. Word) erfolgen, wobei es Ihnen freigestellt ist, entweder eine Ausarbeitung oder eine Präsentation zu erstellen.

In den jeweiligen Aufgaben verwenden Sie eine vorgegebene Anzahl von Datensätze, die Sie bitte dieser Sammlung von Klassifikations- und Regressionsdatensätzen entnehmen, falls die Datensätze nicht in der Aufgabenstellung festgelegt werden. Achten Sie hierbei bitte darauf möglichst unterschiedliche Datensätze zu wählen und diese in den einzelnen Aufgaben zu variieren. Bei der Auswahl der Datensätze ist weiterhin zu beachten, daß bestimmte Lernverfahren nicht mit allen Datensätze umgehen können. Optional können Sie dieses Problem beheben, indem Sie die Daten vorverarbeiten, z.B. mittels `FilteredClassifier` und einem entsprechendem Preprocessing-Filter. In diesem Fall bzw. falls Sie die Standardeinstellungen in Weka modifizieren, geben Sie dies bitte in ihrer Lösung an.

Aufgabe 1 Regellernen: Anwendung und Interpretation

In dieser einführenden Aufgabe sollen Sie die Verwendung von WEKA erlernen und dabei die Ergebnisse dreier Regellerner auf drei unterschiedlichen Datensätzen vergleichen. Wenden Sie hierzu die Regellerner `ConjunctiveRule`, `JRip` und `Prism` auf 3 Klassifikationsdatensätze an.

- Vergleichen Sie die Anzahl der Regeln, der Bedingungen und der vorhergesagten Klassen der resultierenden Regelmengen jeweils in Bezug auf
  - die einzelnen Datensätze
  - die jeweiligen Regellerner
- Existiert bei allen Algorithmen eine Default-Rule? Wenn ja:
  - Welche Klasse wird üblicherweise als Default-Rule ausgewählt?
  - Wie interpretieren Sie die Güte dieser Default-Rule?
- Läßt sich anhand der vorherigen Teilaufgaben eine Aussage treffen, welche der drei Datenmengen am leichtesten bzw. am besten zu lernen ist?
- Vergleichen Sie die Regelmengen der Algorithmen `JRip` und `Prism` für den Datensatz `contact-lenses.arff`. Wie schätzen Sie die Allgemeinheit der von `JRip` bzw. `Prism` gefundenen Regeln ein? Beachten Sie hierbei, daß `JRip` als Heuristik Information Gain und `Prism` Precision verwendet.

Aufgabe 2 Evaluation von Regellernern

In dieser Aufgabe sollen unterschiedliche Evaluierungsmethoden unter Verwendung von WEKA eingesetzt und deren Ergebnisse diskutiert werden. Wenden Sie den Regellerner `JRip` auf 5 Datensätze an. Teilen Sie hierzu jeden Datensatz zunächst in 2 gleich große, stratifizierte Teile, einer Trainingsmenge und Validierungsmenge, auf.

a) Trainieren Sie nun `JRip` auf jeder dieser Trainingsmengen (ggf. auf Teilen dieser Mengen, siehe Cross-Validation usw.) und evaluieren Sie die Genauigkeit (prozentualer Anteil der korrekt klassifizierten Beispiele) der resultierenden Klassifizierer jeweils mittels:
   - 1x5 Cross-Validation

- 1x10 Cross-Validation
- 1x20 Cross-Validation
- Leave-One-Out
- bzw. auf der Trainingsmenge

Wie schätzen Sie die Qualität der erhaltenen Genauigkeitsabschätzungen ein?

Anmerkung: *In dieser Teilaufgabe sollen vorerst keine Veränderungen an weiterführenden Einstellungen, wie z.B. andere Random-Seeds, vorgenommen werden.*

b) Wiederholen Sie Aufgabe a) mit dem Unterschied, daß Sie nun eine 10x10 Cross-Validation zur Evaluation verwenden sollen. Wenden Sie hierzu zehnmal eine 1x10 Cross-Validation mit 10 unterschiedlichen Random-Seeds an und mitteln die erzielten Genauigkeiten.

Vergleichen Sie die so erzielte Genauigkeitsabschätzung mit den Abschätzungen aus der Aufgabe a).

Führt ihrer Meinung nach eine geschickte Auswahl von Random-Seeds zu einer besseren Abschätzung?

c) Bestimmen Sie die Genauigkeit auf der Validierungsmenge (d.h. verwenden Sie diese als Testmenge). Wie schätzen Sie nun unter Annahme, daß es sich bei der Validerungsmenge um einen realen Anwendungsfall handelt, die Abschätzungen der Evaluierungsmethoden aus den Aufgaben a) und b) ein?

Aufgabe 3 ROC-Kurven

- Vergleichen Sie für einen ausgewählten Klassifikationsdatensatz die ROC-Kurven bzw. die Fläche unter diesen Kurven für die Klassifizierer `J48` und `NaiveBayes`. Sie können die ROC-Kurven betrachten, indem Sie mit der rechten Maustaste im Fenster "Result List" den Menü-Punkt "Threshold List" auswählen.
- Interpretieren Sie die Resultate. Sie können die Werte, die zum Zeichnen der Kurve verwendet wurden, auch mit "Save" in ein ARFF-File exportieren, und dieses (nach Löschen des Headers) in Grafik-Programme importieren. So können Sie z.B. beide Kurven (für `J48` und `NaiveBayes`) übereinander legen.

Aufgabe 4 Entscheidungsbäume

- Wählen Sie 2 Klassifikationsdatensätze aus. Vergleichen Sie für diese Datensätze die ROC-Kurven bzw. die Fläche unter diesen Kurven für die Klassifizierer `J48` einmal mit und einmal ohne Pruning (Option 'unpruned') und `ID3`. Bei `J48` verwenden Sie für die anderen Optionen die Default-Werte.
- Vergleichen Sie die Klassifizierer ebenfalls mit den Accuracy Werten der Cross-Validation.
- Betrachten Sie auch die Größe der entstandenen Bäume (Anzahl Knoten und/oder Blätter im Baum) und setzen Sie diese in Zusammenhang mit der Güte der Klassifizierer.

Aufgabe 5 Nearest Neighbour

- Verwenden Sie für diese Aufgabe die gleichen Datensets wie in der vorherigen Aufgabe. Finden Sie heraus, für welches $k \in \{1, 3, 5, 7, 9, 11\}$ der Algorithmus `k-NN` (in `weka` heisst der Algorithmus `IBk`; verwenden Sie auch hier die Default-Optionen) die höchste Cross Validation Accuracy bekommt. Ist der Algorithmus für diesen Wert von k besser als die Entscheidungsbäume der vorherigen Aufgabe?

Aufgabe 6 Regressionsbäume

Benutzen Sie die 5 Regressionsdatensätze für diese Aufgabe (außer dem Datensatz `regression`). Für nominale Attribute beachten Sie bitte, dass der Lerner `M5P` eine Binarisierung der Daten vornimmt ($A = a <= 0.5$ bedeutet also: alle Instanzen, für die A *nicht* den Wert a hat). Die Gesamtanzahl der Instanzen ist n, der tatsächliche Wert einer Instanz j ist y_j und der vorhergesagte Wert einer Instanz j ist r_j (genau wie im Skript).

- Vergleichen Sie den *Mean Absolute Error* ($\frac{1}{n} \cdot \sum_j |y_j - r_j|$) und den *Root Mean Squared Error* ($\sqrt{Mean\ Squared\ Error}$) (10 CV oder Test Set wenn verfügbar), sowie die Modelle (Interpretierbarkeit/Größe) jeweils für den Regressionsbaumlerner `M5P`, einmal mit angeschaltetem Pruning und einmal ohne Pruning (Benutzen Sie Regressionsbäume, also setzen Sie die Option 'buildRegressionTree' auf 'True'). Bringt Pruning bei Regressionstasks eine Verbesserung?

- Verwenden Sie nun Model Trees (Option 'buildRegressionTree' auf 'False' setzen, ansonsten Default Optionen). Vergleichen Sie die Model Trees mit den Regressionsbäumen.
- Verwenden Sie nun den Datensatz `regression`. Dieser entspricht dem Datensatz aus der Übung. Vergleichen Sie den Baum aus der 10. Übung mit einem Regressionsbaum, den Sie mit `M5P` gelernt haben. Verwenden Sie hier einen Regressionsbaum ohne Pruning, der min. 1 Instanz pro Blatt besitzen muss. Betrachten Sie wieder die Größe und z.B. den *Mean Absolute Error* jeweils auf dem Testset (`regression_test`).

Aufgabe 7 Ensemble-Lernen

In dieser Aufgabe sollen unterschiedliche Ensemble-Methoden eingesetzt und deren Ergebnisse verglichen werden. Der Entscheidungsbaumlerner J48 soll als Basislerner verwendet werden. Wählen Sie für diese Aufgabe bitte 5 Klassifikationsdatensätze aus.

a) Bestimmen Sie die Genauigkeit des regulären J48.

b) Verwenden Sie nun Bagging mit J48 und AdaBoost mit J48. Benutzen Sie außerdem noch Random Forests. Bestimmen Sie für die so erhaltenen Klassifizierer die Genauigkeiten für eine stetig wachsende Anzahl von Iterationen (bei den Random Forest verändern Sie bitte die Anzahl der Bäume). Wie interpretieren Sie die Entwicklung der erzielten Genauigkeiten?

Aufgabe 8 Entdecken von Assoziationsregeln

Das Datenset adult.arff enthält Daten von 48.842 US Bürgern über Geschlecht, Ausbildung, Familienstand, Beruf, Einkommen (class Variable), etc. Versuchen Sie, mit dem Apriori-Algorithmus aus Weka in diesem Datenset *interessante* Regeln zu finden. Sie können dabei sowohl die Optionen von Weka ausprobieren (z.B. `-T` das Maß, nach dem die Regeln sortiert werden) als auch das Datenset verändern (z.B. durch Entfernen einzelner Attribute). Beachten Sie, daß in der Version zum Download zwei numerische Attribute enthalten sind, die Sie diskretisieren oder einfach entfernen können. Falls die Laufzeiten zu lange werden (mehrere Minuten), können Sie auch auf einer Teilmenge der Daten arbeiten.

Aufgabe 9 Pre-Processing

Wählen Sie drei Klassifikationsdatensätze aus. Erstellen Sie für jeden Datenset eine diskretisierte Version unter Verwendung des Filters `weka.filters.supervised.attribute.Discretize`.

a) Schätzen Sie die Genauigkeit von `J48` mittels Cross-validation auf den ursprünglichen Daten und auf den diskretisierten Daten ab.

b) Der Meta-Classifier `FilteredClassifier` erlaubt, eine Kombination einer Pre-processing Methode und eines Classifiers zu einem neuen Classifier zu machen. Erzeugen Sie die Kombination `Discretize` und `J48` und schätzen Sie deren Genauigkeit auf den ursprünglichen Daten ab.

Wie interpretieren Sie den Vergleich der Genauigkeiten und der Größe der gelernten Bäume dieser drei Experimente (die Ergebnisse können über die drei Datensets gemittelt werden)?