
Praktikum

Datamining zur automatischen Spielanalyse im Fußball

Features und Entscheidungsbäume 2

Im letzten Blatt haben Sie hauptsächlich vorgegebene Features verwendet um die Merkmale zum Lernen des Entscheidungsbaums zu berechnen. Heute sollen Sie ihre eigenen Merkmale definieren.

Zur Implementierung eines Features müssen die Ein- und Ausgabesignaturen, die von `get{Input|Output}Signature` zurückgeliefert werden festgelegt und die Methode `calculate` implementiert werden.

`QueryFeature` implementiert das Interface `IParameterized<Object>`, das für eine einheitliche Parameterstruktur verantwortlich ist. Damit können etwaige Parameter an die Features übergeben werden. Realisiert wird dies in `QueryFeature` über das Attribut `parameterBase` von der Klasse `ParameterBase`, die alle nötigen Methoden zur Verfügung stellt. Diesem können Sie via `setNecessaryParameters` die für das Feature notwendigen Parameter übergeben und etwaige Standardparameter mit `setParameter[s]` setzen.

Aufgaben:

1. Schreiben Sie ihr eigenes Feature (von `QueryFeature` abgeleitet), das für einen Punkt auf dem Spielfeld ein Potential berechnen kann. Nehmen Sie hierzu für jeden Spieler (x_n, y_n) eine Gaussdichte (Gleichung 1) an, deren Parameter $\sigma_{x/y}$ und N für beide Mannschaften gewählt werden können:

$$P_n(x, y; N, \sigma) = \frac{N}{2\pi\sigma_x\sigma_y} \exp\left(-\frac{(x - x_n)^2}{2\sigma_x^2} - \frac{(y - y_n)^2}{2\sigma_y^2}\right) \quad (1)$$

Definieren Sie sich dazu entsprechende Ein- und Ausgabesignaturen. Die Eingabespalten sollten das Team oder die Richtung des Angriffs, die Position aller Spieler und die Position des zu berechnenden Punktes enthalten. Als Ausgabe sollte die Koordinate des zu berechnenden Punktes, der berechnete Wert und die entsprechende Gruppe zurückgeliefert werden um die Gruppierung für nachgeschaltete Features/Visualisierungen zu erleichtern. Überlegen Sie sich wie die Gruppierung für ihr Feature aussehen muss.

Bauen Sie die Parameter $\sigma_{x/y}$ und N in ihr Feature ein und legen Sie geeignete Standardwerte fest (das Potential für die Angreifer positiv, für die Verteidiger negativ).

2. Wenn man eine Visualisierung für dieses Feature bauen will, muss man für jeden Punkt auf dem Spielfeld das Potential berechnen. Dazu ist der mehrmalige Aufruf

des Features sehr langsam. Machen sie deshalb die Übergabe des Punktes optional und berechnen Sie für das gesamte Spielfeld in einer parametrisierbaren Unterteilung das Potential und liefern sie dann für jeden Punkt das Potential zurück.

3. Schreiben Sie eine Visualisierung, die für einen bestimmten Zeitpunkt die Potentialverteilung auf einem Spielfeld anzeigt.

Schreiben Sie zudem ein Skript, das die beiden Teile (Feature und Visualisierung) zusammenfügt und anzeigt.

4. (optional) Berechnen Sie mit einem weiteren Feature das Integral des Potentials zwischen zwei Punkten. Dies kann z.B. durch Aufteilung der Strecke in geeignete Teile und Aufsummieren der Teilintegrale erfolgen, wobei man annehmen darf, dass das Potential auf den Teilstrecken konstant ist.
5. Verwenden Sie nun zum Lernen eines Entscheidungsbaums zusätzlich das berechnete Potential am Ort des Balls und das Integral des Potentials über die Strecke von Ball zum Tor (optional).
6. (optional) Das Potential am Ort des Balls wird logischerweise vom Potential des ballführenden Spielers dominiert. Übergeben Sie dem Feature statt der/zusätzlich zur Richtung den ballführenden Spieler und schwächen Sie dessen Potential ein wenig ab. Visualisieren Sie das Resultat erneut.
7. **Zusatzaufgabe:** Man kann auch verschiedene andere Klassifikatoren mit WEKA lernen. Dies kann durch `Result→BuildClassifier...` im ContentTree geschehen. Dies öffnet ein aus der WEKA-Oberfläche entnommenes Fenster mit dem Preprocessing, das Lernen von Klassifikatoren, Clusterern usw. durchgeführt werden kann. Machen Sie sich ein wenig mit WEKA vertraut und probieren Sie einige Klassifikatoren für ihre Daten aus.

WEKA kann auch eigenständig ausgeführt werden WEKA kann zum Beispiel mit `java -cp weka_fipm.jar weka.gui.GUIChooser` gestartet werden. Der Explorer ist der auch in FIPM integrierte Teil. Der Austausch der Daten muss dann aber über Dateien erfolgen, die man beispielsweise durch `Result→Export...` generieren und in WEKA einlesen kann.