Intelligente Systeme
Aufgabe 2
Dokumentation/Report

Einleitung:

„ Ausgangssituation und Zielsetzung

Bei dieser Aufgabe geht es um eine Klassifikation von Verhaltensmustern: Anhand von individuellen Bewegungsdaten (von Fischen) soll bestimmt werden, ob das betreffende Individuum alleine war oder sich in einer Gruppe (von Artgenossen) befand. Der Klassifikator soll also nicht Unterschiede zwischen einzelnen Individuen erkennen, sondern individuenunabhängige Unterschiede zwischen zwei Formen von sozialem Kontext.

Für das Training des Klassifikators sind die Dateien *train_alone.txt* und *train_group.txt* zu verwenden und für die Evaluation die Dateien *eval_alone.txt* und *eval_group.txt*. Alle vier Dateien besitzen denselben Aufbau. Jede Zeile enthält Bewegungsdaten für ein einzelnes Individuum, bestehend aus einer Sequenz von Positionen, die durch Semikolons getrennt sind. Jede Position besteht aus einer X- und einer Y-Koordinate, die durch ein Komma getrennt sind. Zwischen zwei Positionen ist stets dieselbe Zeit vergangen.

Die Dateien *train_alone.txt* und *eval_alone.txt* enthalten Daten von Individuen, die während der Beobachtung alleine waren, und die Dateien *train_group.txt* und*eval_group.txt* Daten von Individuen, die während der Beobachtung in einer Gruppe waren.

***Bitte beachten**: Alle Datensätze enthalten **echte** Daten, die nur für diese Lehrveranstaltung verwendet und nicht weitergegeben werden dürfen!*

Konkrete Aufgabenstellung

Für die Lösung ist eine Bayes'sche Klassifikation zu verwenden (s. Skript, Folien 102 -104), bei der die Wahrscheinlichkeit P(b|c) durch eine Markovkette erster Ordnung modelliert wird (s. Skript, Folien 110 -120). Zur Vereinfachung darf davon ausgegangen werden, dass beide Klassen ("Alleine" und "In einer Gruppe") gleichhäufig auftreten.

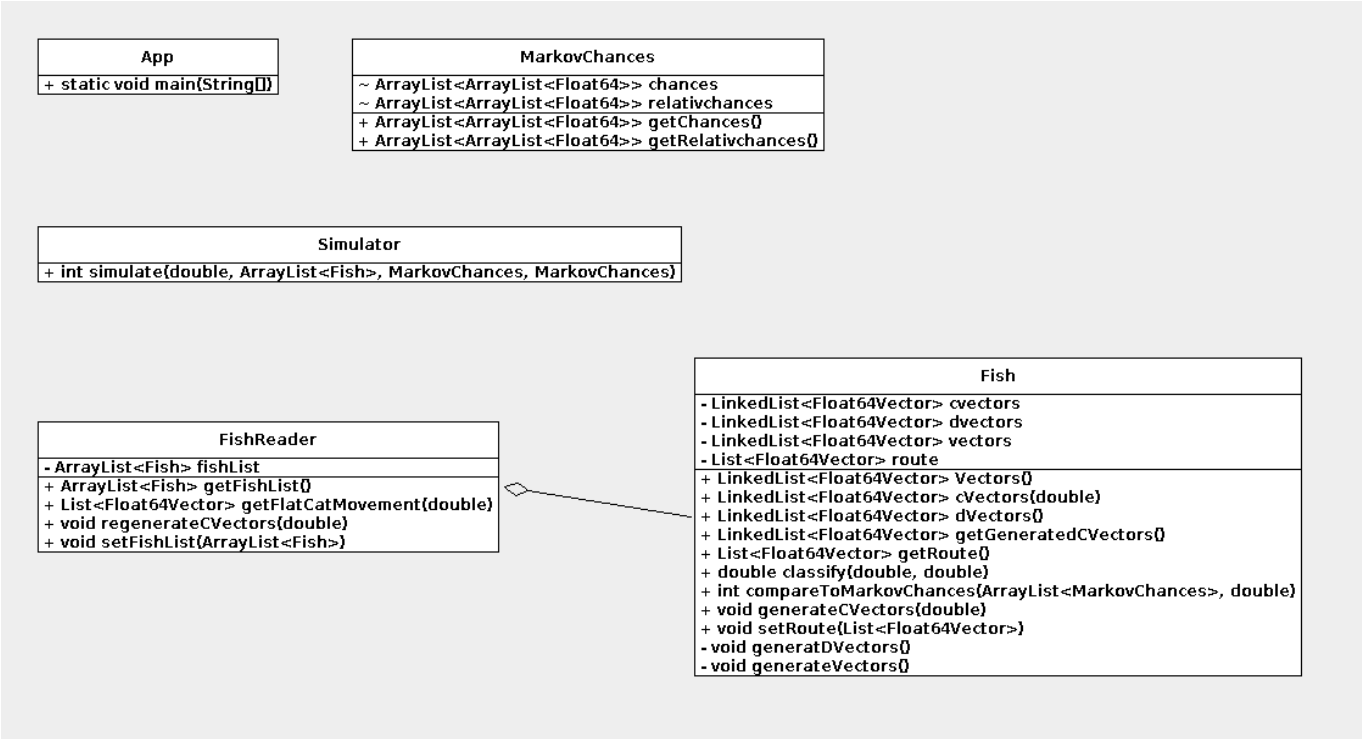
Die Zustände der Markovkette müssen relevante Aspekte der Bewegungsmuster abbilden. Hierfür gilt folgende Vorgabe: Die Zustände sind zu bilden, indem die Änderungen der Geschwindigkeitsvektoren in Kategorien eingeteilt werden. Genauer: Aus je zwei aufeinanderfolgenden Positionen (x_1, y_1) und (x_2, y_2) wird ein Vektor ($x_2 – x_1$, $y_2 – y_1$) gebildet und aus je zwei aufeinanderfolgenden Vektoren (v_{x1}, v_{y1}) und (v_{x2}, v_{y2}) wiederum die Differenz d = ($v_{x2} – v_{x1}$, $v_{y2} – v_{y1}$). Indem die Werte der beiden Komponenten von d in Kategorien eingeteilt werden, kann die unendlich große Zahl von möglichen Werten auf eine endliche Zahl von Zuständen abgebildet werden. Für diese Aufgabe genügt es, drei Kategorien für einen Wert w zu verwenden, die durch einen Schwellwert k definiert sind:

A: $w < –k$
B: $–k \le w \le k$
C: $w > k$

Damit ergeben sich die folgenden 9 möglichen Zustände für einen Differenzvektor d: (A,A), (A,B), (A,C), (B,A), ... (C,C).
Ermitteln Sie für jede der beiden Evaluationsdateien (*eval_alone.txt* und *eval_group.txt*) den Prozentsatz der korrekt klassifizierten Sequenzen in Abhängigkeit vom Schwellwert k. Wählen Sie für k einen sinnvollen Bereich. (Wenn Sie alles richtig machen, sollte es Werte für k geben, mit denen auf beiden Dateien ein Ergebnis über 90% erreicht werden kann.)
Abzugeben sind der Quellcode der Java-Klassen sowie eine kurze schriftliche Darstellung der Ergebnisse (gerne stichwortartig) als PDF-Dokument. “

– Zitat: https://lernraum.fh-luebeck.de/mod/assign/view.php?id=70091

Klassendiagramm:



Algorithmus mit Erklärung:

→ Die Dateien werden im FishReader eingelesen, pro Zeile in der Datei wird ein Fisch erstellt, alle Fische in dieser Datei werden in der fishList im FishReader gespeichert.

→ Die Route des Fisches repräsentiert die aufeinander folgenden Punkte in der Zeile des Fisches.

→ Aus diesen Punkten werden Vektoren erstellt, daraus die Differenzvektoren, deren Komponenten in Klassen (0,1,2 == A,B,C) eingeteilt und in der Liste dVectors im Fisch gespeichert werden.

→ Die Klasse MarcovChances erstellt aus einem FischReader und einem Schwellwert k eine absolute und daraus resultierende relative Häufigkeit jedes möglichen Zustandspaares, die zur Bestimmung der Fische aus den Evaluationslisten herangezogen wird.

->Hierzu werden die klassifizierten Häufigkeiten aller Fische iteriert, anhand der Klassifizierungen des aktuellen und nachfolgenden Paares Koordinaten der jeweiligen Kombination in einer 9*9 Matrix errechnet und die zuvor mit 1 initialisierte Matrix am entsprechenden Punkt inkrementiert. Diese absolute Matrix wird in eine relative Matrix umgerechnet.

→ Die simulate() Methode der Simulatorklasse erwartet einen k Wert, eine Liste mit Fischen, die zu klassifizieren sind und zwei MarcovChances Klassen der Train-Daten(group und alone). Der Wert der richtig klassifizierten Fische wird zurück gegeben.
Die klassifizierten Differenzvektoren der einzelnen Fische werden herangezogen und bilden den Index aus der relativen Häufigkeitstabelle der MarcovChances Klassen. Diese Häufigkeiten werden, je herangezogener Tabelle, miteinander multipliziert (log wird gebildet und addiert) und dann anhand der resultierenden Werte klassifiziert.
Die Klassifikation entspricht der des jeweils größeren zu einer Häufigkeitstabelle gehörenden ausmultiplizierten Wertes.

->Die richtig klassifizierte Anzahl von Fischen wird gezählt und samt zugehörigem k-Wert gespeichert, sollte die Anzahl größer als der zuvor gefundene Wert sein.
Hiernach wird der k-wert inkrementiert und die gesamte Berechnung mit aktuellem k-Wert erneut durchgeführt.

->Nach der eingestellten Anzahl von Iterationen (Aktuell 20) wird das Programm abgebrochen und das beste gefundene Ergebnis und dessen k-Wert ausgegeben.
Dieser k-Wert ist der gefundene optimale k-Wert.

Auswertung:

Die Schleife hat den Startwert 0,1 und den Endwert (loop-Bedingung) 20. Die größe der Iterationsschritten liegt bei 0,1.

Output: „Best k value found at: 13.699999999999967 with matches in alone : 45 (93.75)%
Best k value found at: 13.699999999999967 with matches in group : 44 (91.66666666666666)%“

An dem Diagramm ist zu erkennen, dass der richtige Schwellwert für k bei 13,69 liegt. Bei diesem Schwellwert wird bei beiden Evaluationsdaten ein Wert von über 90% richtiger Klassifizierungen erreicht.