Setup

Die zu analysierende Datei muss im dataset ordner plaziert sein. Danach kann einfach

Aufgabenstellung

Es wird eine CSV-Datei bereitgestellt, welche aus einem Video Gesichts-Charakteristika (bzw. Features) extrahiert. In den ersten vier Feature-Spalten werden die Position des Frames sowie die Größe des Frames angegeben. In den darauffolgenden Spalten werden in folgender Reihenfolge tatsächliche Features im Gesicht des Beobachteten dokumentiert:

* Stirnfalte
* Augenöffnung und Brauenabstand auf jeweils beiden Seiten
* horizontale und vertikale Falten auf der Nasenwurzel
* Faltenbildung auf beiden Wangenseiten
* Mundöffnung

Jede Zeile in der CSV-Datei stellt einen Frame des Videos dar. In der Aufgabenstellung ist eine bestimmte Reihenfolge bzw. Abfolge von Emotionen festgelegt.

Die Aufgabe ist es nun, für jeden Frame der Videosequenz (dem Programm im Format einer CSV-Datei bereitgestellt) mithilfe der ‚Dempster-Shafer-Regel‘ eine Emotion auszugeben, basierend auf den eingelesenen Features. Zwar sollen erst die Emotionen der letzten 25 Frames ausgegeben werden, da der Gruppe aber unbekannt ist wie die Testdaten bei der Bewertung aussehen, wird für jeden Frame die erkannte Emotion ausgegeben. Zusätzlich ist zu beachten, dass bei den Testdaten der Bewertung unvollständige Featurewerte auftreten können.

Theorie

# Mapping: Aufgabenstellung auf Dempster-Shafer-Regel

In dieser Bearbeitung existiert eine Menge von Alternativen Ω , welches die Menge aller möglichen Emotionen beschreibt. Die Alternativenmenge ist durch die Aufgabestellung vollständig gegeben und es ist anzunehmen, dass diese sich gegenseitig ausschließen. Somit sind alle Voraussetzungen für Evidenztheorie erfüllt.

Gehen wir nun für die Emotionsbestimmung von einem Frame aus. Jedes gemessene Feature (ab Spalte 5) wird zur einem Basismaß. Dieses Basismaß besteht immer aus mindestens 2 Einträgen. Ein Eintrag setzt sich zusammen aus einer Menge an Emotionen, auch genannt ‚fokale Menge‘ und deren Konfidenz. Jede Evidenz hat immer ein Ω - Element, also ein Eintrag in der die fokale Menge die gesamte Menge von Alternativen ist.

Schlussendlich werden alle Basismaße, also alle gemessenen Features, akkumuliert. Daraus entsteht das Basismaß mgesamt.

Aus dem Basismaßen in mgesamt kann dann für jede Emotion die Plausibilität errechnet werden. Die Emotion mit der höchsten Plausibilität wird schlussendlich als erkannte Emotion ausgegeben.

# Mapping: Daten zu Kategorisierung mit stuff

@Cathleen: insert stuff about your magic here

Methode

# Umsetzung Algorithmus

Gesteuert wird der gesamte Ablauf des Programms von der Main-Klasse aus.

Input CSV

Output Datei

Main

Dempster-S-Rule

Basismaß

Normalisierung

Entry

Der Ablauf ist in der Grafik dargestellt. Es wird in der Main der Input in Form einer CSV Datei eingelesen. Der Inhalt dieser wird dann in einer Schleife pro Zeile/Frame und zur Normalisierung weitergegeben. Die Normalisierung ist ein langer Prozess und deshalb im nächsten Subkapitel beschrieben.

Zurück kommt für den aktuellen Frame eine Liste; für jedes Feature im Input wird die Fokalmenge sowie die Konfidenz für diese Fokalmenge zurückgegeben. Im nächsten Schritt wird eine Liste von ‚Basis-Basismaßen‘ angelegt, es wird also für jedes Feature ein Basismaß angelegt. Diese Basismaßklasse greift im Constructor auf die Klasse Entry zu. Es wird zuerst ein Eintrag für die gegebene Fokalmenge und Konfizenz angelegt und zu der Entry-List dieser Evidenz hinzugefügt. Folgend daraus wird ebenfalls der Omega-Entrag angelegt mit der Gegenwahrscheinlichkeit bzw. -Konfidenz.

Im nächsten Schritt wird durch diese ‚Basis-Basismaße‘-Liste iteriert und sie werden Schritt für Schritt akkumuliert. Dazu werden jeweils immer zwei Basismaße an die Dempster-Rule Klasse übergeben. Diese greift wiederum nocheinmal auf die Basismaßklasse zu, um als Output auch ein Basismaß zurückgeben zu können. Dort wird dann tabellenartig jeder Eintrag miteinander verrechnet. Gegegeben seien dazu die Basismaße mk und mj, die akkumuliert werden sollen zum Output-Basismaß mo. Die DSR-Klasse unterscheidet dazu folgende Fälle:

1. Eines der Einträge ist Omega:   
   Es wird in mo ein Eintrag angelegt mit der Konfidenz bestimmt durch das Produkt der Konfidenzen von mk und mj und der fokalen Menge gleich der fokalen Menge des Elements das nicht Omega ist.
2. Beide Einträge sind Omega:  
   Es wird in dem Output-Basismaß ein Eintrag angelegt mit der Konfidenz bestimmt durch das Produkt der beiden Input-Einträge und der fokalen Menge Omega
3. Keines der Einträge ist Omega:  
   Hier werden zuerst die beiden fokalen Mengen miteinander verglichen und die Schnittmenge betrachtet
   1. Die Schnittmenge ist leer  
      Es wird trotzdem ein Eintrag in dem Output-Basismaß mit entsprechender Wahrscheinlichkeit angelegt, dies wird später separat bearbeitet
   2. Die Schnittmenge ist nicht leer

Es wird in dem Output-Basismaß ein Eintrag angelegt mit der Konfidenz bestimmt durch das Produkt der beiden Input-Einträge und der fokalen Menge gleich der bestimmten Schnittmenge

In jedem dieser Schritte wird vor dem Erstellen des Eintrags sicher gestellt, dass ein Eintrag mit derselben fokalen Menge nicht schon in dem Output-Basismaß existiert. Ist dies der Fall, wird die ausgerechnete Konfidenz des aktuellen Eintrags der Konfidenz des bereits existierenden Eintrags hinzuaddiert. Die einzige Ausnahme, in der dieser Vergleich nicht durchgeführt wird ist der Fall, in dem beide Input-Einträge Omega sind, da dieser Fall per Definition nur einmal pro Akkumulation vorkommt.

Als letzter Schritt bevor das Output-Basismaß zurückgegeben werden kann werden die Einträge mit leeren fokalen Mengen bereinigt. Dazu wird die Konfidenz dieses Eintrags bestimmt und dann anhand des Faktors Fkorrektur = die Konfidenzen der restlichen Einträge neu gesetzt.

Es ist zu beachten, dass nach der Neuberechnung nochmal ein Kontrollcheck implementiert wurde, der überprüft, ob die Konfidenzen addiert noch immer 1 ergeben.

Bei dem Kontrollcheck wie auch bei den Tests, die zur Kontrolle der Funktionen implementiert wurden, mussten wir leider auf Runden zurückgreifen. Da die Konfidenzen als float implementiert wurden kam es zu Rundungsfehlern bei dem Ergebnis. Wir haben uns deshalb für den Check und die Tests dafür entschieden die zu vergleichenden Ergebnisse auf 5 Nachkommastellen zu runden.

Wurde das Basismaß mgesamt aus allen vorhergehenden Basismaßen berechnet, wird auf dieses Basismaß die Funktion calculate\_plausibility() aufgerufen. Diese gibt eine Liste aller vorhandenen Emotionen in mgesamt  aus, sowie ihre Plausibilität. In der Main Klasse wird dann die Emotion mit der höchsten Plausibilität als Output gewählt und zusammen mit ihrer Plausibilität in die Output-Datei geschrieben.

# Umsetzung Daten-Normalisierung

Wieso wir bei x is omega and y is omega kein addieren brauchen

Ergebnisse

Future Work

Bei der Bearbeitung der Aufgabenstellung gab es in der Gruppe zwei Vorschläge für mögliche Erweiterungen, die aufgrund der Zeit leider nicht mehr eingebunden werden konnten. Wir möchten dies aber trotzdem nennen, da wir denken, dass diese Erweiterungen die Analyse nochmals verbessern könnten.

**Kontext bei real-world Anwendung**

Für diese Bearbeitung haben wir uns dazu entschieden, für jeden Frame die erkannte Emotion auszugeben. Laut der Aufgabenstellung ändert sich (zumindest am Anfang) in bestimmten Abständen die Emotion. Nun ist es nicht unwahrscheinlich, dass die Emotion nicht durchgehend klar zu erkennen ist oder die Testperson beim Wechseln des Gesichtsausdrucks keine ‚klare‘ Emotion zeigen kann. In einem realen Anwendungsszenario wäre es also sinnvoll – wenn man die Abfolge und Länge der Emotionen kennt – die Analyse eines Frames mit Kontext zu betrachten. Folgendes Outputbeispiel verdeutlicht dies (es ist bekannt, dass eine Emotion für 5 Sekunden gehalten wird):

|  |  |
| --- | --- |
| Fear – 0.97  Fear – 0.97  Fear – 0.97  Fear – 0.97  Neutral – 0.56 | Joy – 0.82  Joy – 0.82  Joy – 0.82  Sadness – 0.60  Joy – 0.82 |

Hier ist deutlich zu sehen, dass man leicht Fehlinterpretationen eliminieren kann, in dem man bei einem oder zwei Outliern diese ‚ignoriert‘ bzw. nur eine Emotion pro 5 Sekunden ausgibt.

**K-Means Clustering**

@Cathleen: insert your stuff here