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 (OMEGA), 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 für einen Frame aus. Jedes gemessene Feature (ab Spalte 5) wird zur einer Evidenz. Diese Evidenz besteht immer aus mindestens 2 Basismaßen. Ein Basismaß setzt sich zusammen aus einer Menge an Emotionen, auch genannt ‚fokale Menge‘ und deren Konfidenz. Jede Evidenz hat immer ein ‚Omega‘-Element, also ein Basismaß in der die fokale Menge die gesamte Menge von Alternativen ist.

Schlussendlich werden alle Evidenzen, also alle gemessenen Features, akkumuliert. Daraus entsteht die Evidenz (m\_gesamt).

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

Was ist mit Belief????

# 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

Evidenz

Normalisierung

Basismaß

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-Evidenzen‘ angelegt, es wird also für jedes Feature eine Evidenz angelegt. Diese Evidenzklasse greift im Constructor auf die Klasse Basismaß zu. Es wird zuerst ein Basismaß für die gegebene Fokalmenge und Konfizenz angelegt und zu der Entry-List dieser Evidenz hinzugefügt. Folgend daraus wird ebenfalls das Omega-Basismaß angelegt mit der ??invertierten Summe?.

Im nächsten Schritt wird durch diese ‚Basis Evidenzen‘-Liste iteriert und sie werden Schritt für Schritt akkumuliert. Dazu werden jeweils immer zwei Evidenzen an die Dempster-Rule Klasse übergeben. Diese greift wiederum nocheinmal auf die Evidenzklasse zu, um als Output auch eine Evidenz zurückgeben zu können. Dort wird dann tabellenartig jedes Basismaß miteinander verrechnet. Die DSR-Klasse unterscheidet dazu folgende Fälle:

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

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

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

Als letzter Schritt bevor die Output-Evidenz zurückgegeben werden kann werden die Basismaße mit leeren fokalen Mengen bereinigt. Dazu wird die Konfidenz dieser Basismaße definiert und dann anhand des Faktors 1/1-k die Konfidenzen der restlichen Basismaße neu definiert.

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 nun die letzte akkumulierte Evidenz aus allen vorhergehenden Evidenzen berechnet, wird auf diese Evidenz die Funktion calculate\_plausibility() aufgerufen. Diese gibt eine Liste aller vorhandenen Emotionen in dieser Evidenz 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.

**K-Means Clustering**

@Cathleen: insert your stuff here