Machine Learning CBC Assignment 4 Report

Aadil Khan, Gareth Jones, George Eracleous, Valdas Kriauciukas
Email: {aak08, gdj08, ge108, vk308}@imperial.ac.uk
Group Number: 23
Tutorial Helper: Brais Martinez

Table of Contents

1 Implementation details

1.1 CBR Structure

The cases within the CBR system follow a structure divided into three fields known, individually, as problem, solution and typicality. The problem field is a list holding all the action units (AUs) that are active from a particular example in the training set. The solution field holds the corresponding emotion label attached to the problem, either initialised from the training set or handled by the rest of the CBR system. The typicality field represents the number of occurrences of any particular problem appearing in the CBR system.

The CBR system stores the cases into six groups corresponding to each of the six emotions. These newly found cases are then organised on the basis of which emotion they are labelled as. For each emotion in this system comes a structure holding the label to identify the group (emotion) being handled, the list of associated cases, and a list called *index*. The *index* holds the active AUs that appear throughout the entirety of the group (the emotion). Via assessment of *index,* gives the advantage of checking for similar cases, cases that may or may not belong to a group because all AUs in *index* come from the joint contribution of all cases in the group.

1.2 Retrieve

The function *retrieve* attempts to find the most similar case in the CBR system to that of the new case. In turn, the AUs of the new case's problem field are checked against the *index* list of all groups of the CBR system. If the AU is found to be present in any of the groups, the cases of that group are added to a list for further evaluation. Once all the relevant cases are finally gathered, the new case is compared against the retrieved list of cases. The objective now is to find the most similar case to the new case.

A heuristic (an accumulation of typicality, size and a similarity result) is used here to help make the judgement for most similar case. A case has high similarity to the new case if its typicality value is high, the more times a case presents itself then the higher the chance of it being similar to the new case. Similarity is increased even further if the size of the case is large enough. The intuition behind this is that a large case would embody more AUs and thus be a probable candidate to being similar to another case. The last to contribute to the heuristic is the similarity value where a methodical approach to measure the similarity between a new case and another case is preformed. However, with regards to the heuristic, the larger this value then the higher the chance of the two cases being similar. More on the different types of similarity measures is discussed below.

Ultimately, the case with the highest heuristic is deemed most similar to the new case and this case, the case from the CBR system, is passed on to *reuse* where a solution can be formulated for the system.

1.3 Reuse

In *reuse,* the solution of the most similar case (as returned by *retrieve*) is given to the new case being handled. The result is a 'solved case' holding the problem description of a new case and the solution of an already present case. This 'solved case' is then given to *retain* to assess its occurrence in the CBR system.

1.4 Retain

The function *retain* adds a case to its associated group. If the case to be placed into the system is a new case then it is placed into the appropriate group by simply assessing the solution of the case, the group is now updated. The index of this group is also updated to reflect all the active AUs that are contributed by the new case yet not in *index.* However,if the case is already in the CBR system, all that is needed is to increment the *typicality* and to copy over the solution if currently have none.

2 Similarity measures

We have implemented several similarity measures and more specifically, four of them: a naïve method based on vector lengths, the Euclidean and Manhatan measures and the Jaccard coefficient. Below we explain how these measures are calculated and their relative merits. The similarity measures are used when we try to find the best matching case for a new case. The closer a new case is to an existing case the better.

2.1 Naive method

This measure is based on the assumption that similar emotions will have a similar number of AU's activated. Therefore, the similarity is the absolute value of the difference between the length of the two problem vectors.

2.2 Euclidean

This distance measure is also known as L2-norm. The calculation of the Euclidean distance is based on the Pythagorean theorem and it finds the minimum distance between two points in the vector space. For each index in the vectors we calculate the difference between the corresponding elements of the two vectors and then the difference is raised to the power of two. Finally the square root of the sum of the squared differences gives the Euclidean distance between the two vectors.

2.3 Manhattan

Another distance measure we have implemented is the Manhatan distance (L1-norm). This measure calculates the distance between two points measured along axes at right angles, just like travelling in a city's blocks. More specifically, it sums the absolute value of the differences between each column of the problem vectors.

2.4 Jaccard

The Jaccard coefficient measures the similarity between samples. It is defined as the ratio of the size of intersection to the size of the union of the two samples. Therefore, if the intersection and the union of two sets have the same length then this means that the two vectors are the effectively same and the Jaccard coefficient is 1. In our case the two samples are the problem vectors for each case.

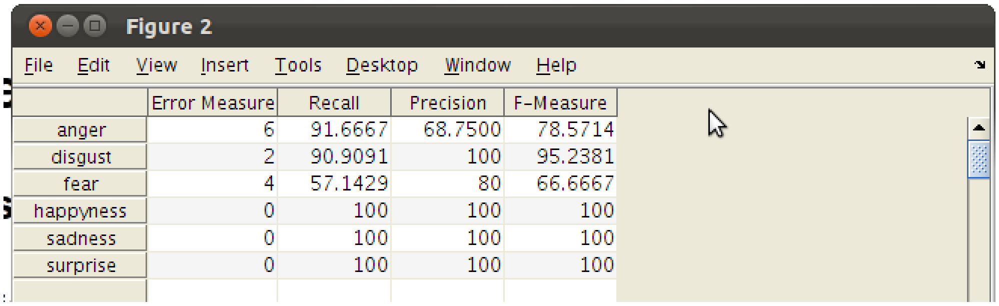
In the next section we present the results using different similarity measures and we explain why some of them are better than the others.

3 Results

3.1 Distance measures comparison

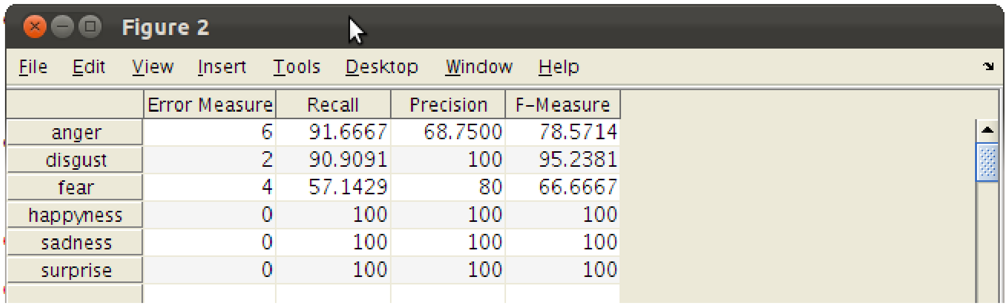
The tables below show the average results for 10-fold cross validation for each distance measure.

Euclidean distance



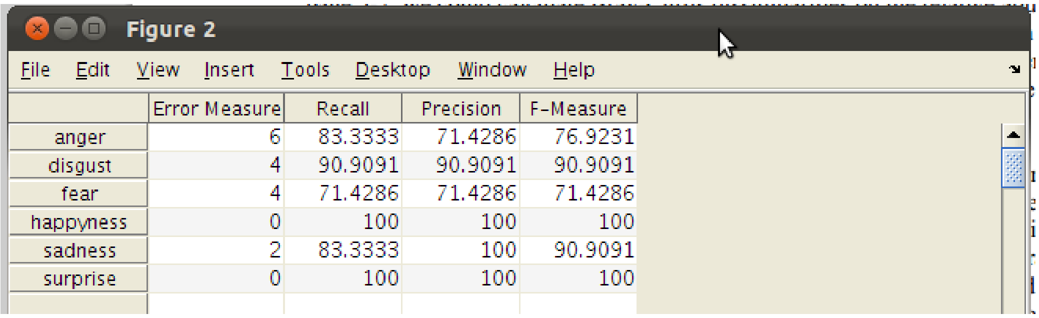
	Error Measure	Recall	Precision	F-Measure
anger	6	91.6667	68.7500	78.5714
disgust	2	90.9091	100	95.2381
fear	4	57.1429	80	66.6667
happyness	0	100	100	100
sadness	0	100	100	100
surprise	0	100	100	100

Manhatan distance



	Error Measure	Recall	Precision	F-Measure
anger	6	91.6667	68.7500	78.5714
disgust	2	90.9091	100	95.2381
fear	4	57.1429	80	66.6667
happyness	0	100	100	100
sadness	0	100	100	100
surprise	0	100	100	100

Jaccard coefficient



	Error Measure	Recall	Precision	F-Measure
anger	6	83.3333	71.4286	76.9231
disgust	4	90.9091	90.9091	90.9091
fear	4	71.4286	71.4286	71.4286
happyness	0	100	100	100
sadness	2	83.3333	100	90.9091
surprise	0	100	100	100

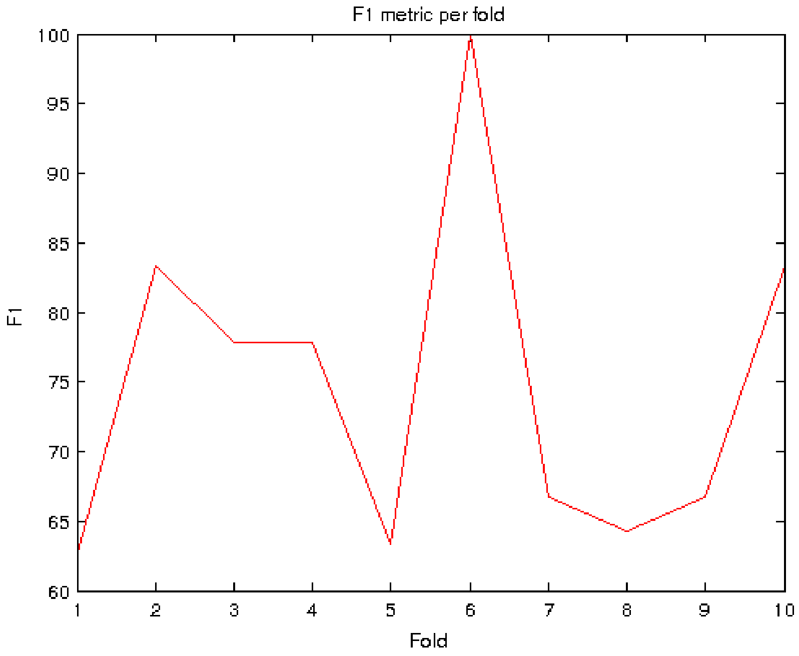
The tables above are the average results obtained with 10-fold cross validation for the different similarity measures. Note that we didn’t include the results of the naive method since they were significantly different than the results of the other methods. As we expected the naive method gave us very poor results compared to the other methods. An explanation for this is that this method made the assumption that similar emotions have problem vectors with similar lengths. However, this might not be true.

Another important observation is that the results obtained with the Manhatan distance and the Euclidean distance are identical. The reason might be that the two measures rely on the same principle for calculating the distances and the AU expressions are binary. Therefore, the individual differences will be either 0 or 1 and the formula for the Euclidean distance reduces to the square root of the Manhatan distance.

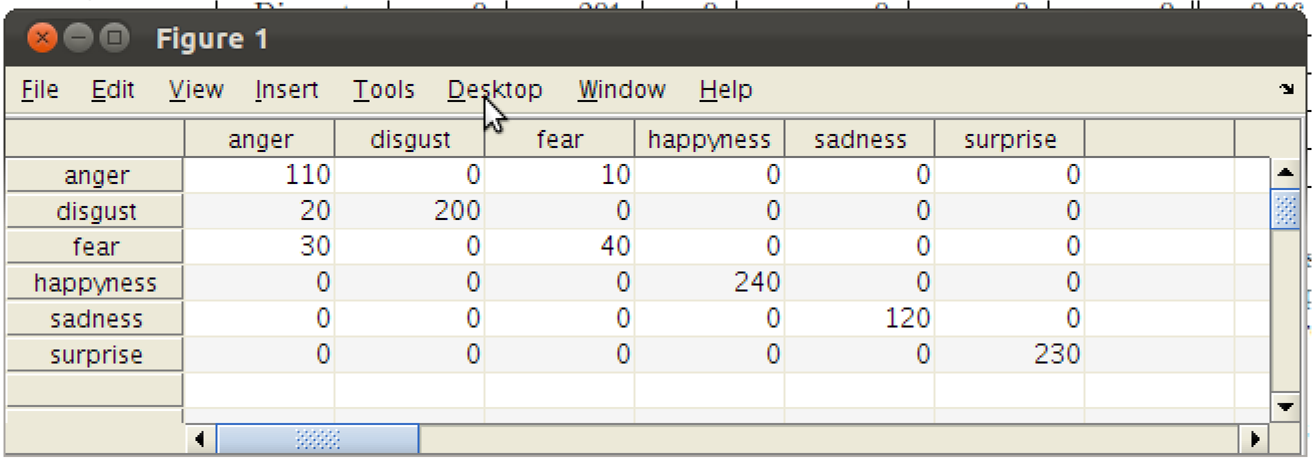
The Jaccard coefficient produces worse results than the two other methods. The Euclidean and Manhatan distances appear to work better than the Jaccard coefficient because they do a point by point comparison on the two vectors and use it to compute the minimum distance between them. In the rest of this section we will use the Euclidean measure to produce our results.

4 Classification scores per fold

F-measure per fold



Average confusion matrix for 10-fold cross validation



	anger	disgust	fear	happyness	sadness	surprise
anger	110	0	10	0	0	0
disgust	20	200	0	0	0	0
fear	30	0	40	0	0	0
happyness	0	0	0	240	0	0
sadness	0	0	0	0	120	0
surprise	0	0	0	0	0	230