Group Name: Deus Ex Machina (Undergraduate)

Yukun Huang	V00804445	kennethhyk@gmail.com
Cameron Mulgrew-MacFarlane	V00775582	Cammac93@gmail.com
Willy Su Yep	V00795480	Op4977377832@gmail.com
Sam Wheating	V00816465	SamWheating@yahoo.ca
Matt Johnson	V00757521	Matthewj@uvic.ca

CENG 420 Assignment #3

QUESTION 1) *[CENG-420: 80 Points] [ELEC-569A: 100 Points]*
K-Nearest Neighbors Algorithm is a supervised machine learning algorithm that we can use for classification and regression problem. By default, the KNN works for binary classification problems. Work with your team to extend the KNN to work with one-class classification problem where the training data the algorithm has one class, and in the production, the algorithm works with more that one class.

The KNN algorithm can be modified to function as a unary classifier by requiring a certain number of positive neighbours (test data samples) within a given radius / margin of error / similarity. This will work to identify positive and negative matches without needing a population of negative examples as the KNN typically does. (See Figure 1 below).

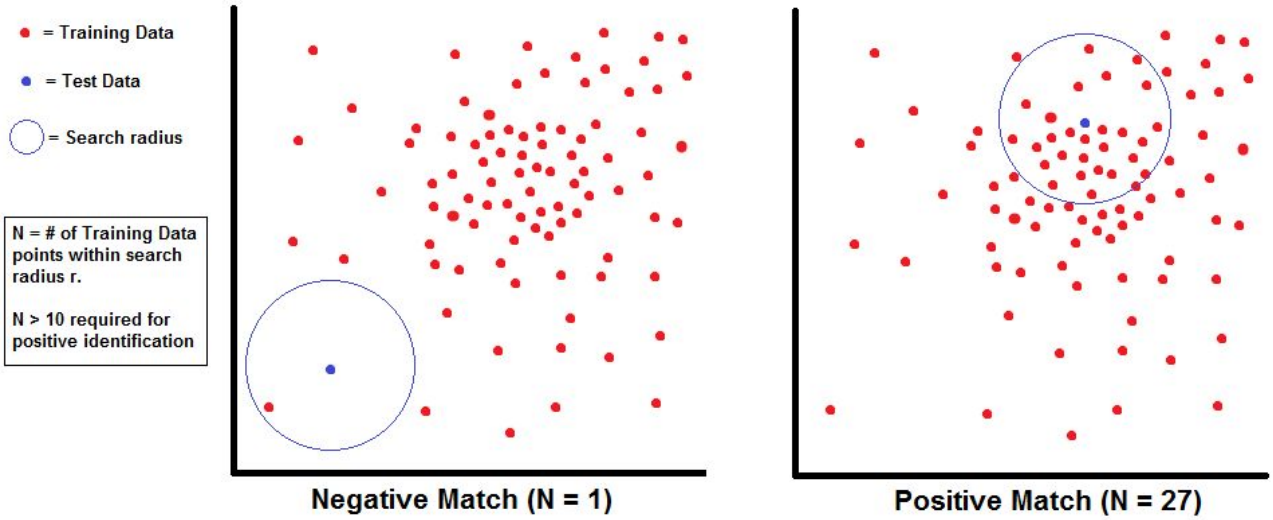

Figure 1: K-Near-Neighbours Unary Classifier. Both r and N can be adjusted / tuned in the validation process. .

This algorithm was implemented in python 3.5, using Euclidean distance as a measure of similarity. Through an iterative validation process it was found that requiring 10 training samples with a euclidean distance less than 2.0 resulted in the highest classification accuracy with the given data. This is the learning portion of this algorithm, as it can run validation tests on a wide range of both N and r until the optimal combination is found.

Using these constants, the unary classifier could classify irises as Iris Setosa or 'Not Iris Setosa' with ~99% accuracy. This was calculated by running ten tests, each using 15 random Iris Setosa samples as training data and a mix of 5 (different) Iris Setosa and 5 Iris Versicolor samples as test data. With additional feature engineering and algorithm optimization (such as weighting the distance function to favour certain features) it may be possible to get a higher classification accuracy.

Overfitting should not be a concern with this algorithm as it relies on the relative position of ten other samples and thus there is an averaging effect which can reduce the potential for overfitting. With a very small value of N, this could become a problem. For example, with N=1, a given non-iris setosa sample could be an outlier from its own set and match closely to a single sample despite not being of the same class. However, Having a larger value of N prevents this.

Please see the attached python code **UnaryIris.py**. The Iris data was imported through SciKit Learn, and the algorithm and testing was implemented manually using standard functions as well as the random, math and numpy libraries.

Pseudocode:

Note: Optimized values found to be N = 10, r = 2.

```
TrainingData = 15 Random Samples from Iris Setosa
TestData = 5 Random Iris Setosa and 5 Random Iris Versicolor

For TestSample in TestData
        Count = 0
        For TrainSample in TrainingData:
                If EuclideanDistance(TestSample, TrainSample) < r:
                        Count += 1
        If Count > N:
                TestSample is Iris Setosa
        Else:
                TestSample is not Iris Setosa
```

QUESTION 2) *[CENG-420: 70 Points] [ELEC-569A: 50 Points]*
The database below is from Movies Night dataset. Each row has a collection of movies watched by a group of users. What association rules can be found in this set if the target minimum support (i.e coverage) is 60% and the target minimum confidence (i.e. accuracy) is 80% ?

T1:KingArthur, AmericanPie, Daredevil, BatmanvsSuperman
T2:Cinderella, AmericanPie, BatmanvsSuperman, Enchanted,
T3:Daredevil, AmericanPie, Cinderella, Enchanted, BatmanvsSuperman
T4:BatmanvsSuperman, AmericanPie, Daredevil

$$Rule: X \Rightarrow Y$$
$$Support(X,Y) = \frac{frq(X,Y)}{N}$$
$$Confidence(X,Y) = \frac{frq(X,Y)}{frq(X)}$$

Potential Rules:

The following rules were found through the Apriori Algorithm:

If a group watches Daredevil and Batman Vs Superman, They will also watch American Pie. (Support = 75%, Confidence = 100%)

If a group watches American Pie and Daredevil, they will also watch Batman Vs Superman. (Support = 75%, Confidence = 100%)

If a group watches Daredevil, they will also watch American Pie and Batman vs. Superman. (Support = 75%, Confidence = 100%)

The following additional rules were found through simple analysis of the data set:

If a group doesn't watch King Arthur, they will watch American Pie and Batman Vs Superman. (Support = 75%, Confidence = 100%)

If a group watches Batman vs. Superman, they will watch American Pie (and Vice-Versa). (Support = 100%, Confidence = 100%)

Note: See Attached Spreadsheet for the Apriori Algorithm steps.

QUESTION 3) *Design a test Question Involving one of the following subjects:*

- *Supervised Learning*
- *Unsupervised Learning*
- *Neural Network and Deep Learning*

Explain how the value of K used in the K-nearest neighbours algorithm affects the decision making process and contributes to overfitting and underfitting. Is a higher, or lower value of K more desirable?