

**Case Study 3： Visualisation in machine learning**

**Name: Omar Jabri ID: 2519359J**

**Name: Yifan Xie ID: 2509996X**

**Name: Tomas Simutis ID: 2519359J**

**Introduction**

* Machine learning uses data to make predictions. Building successful machine learning software necessitates a certain degree of intuition about the dataset's modelling. Exploratory visualisation may be used to encode elements of a dataset for the human cognitive system by representing the structure of the dataset.
* In this case study, we will examine techniques that use **unsupervised machine learning** to classify data structures that can be used to obtain a deeper understanding of a dataset. Clustering and dimensional reduction are two typical unsupervised learning activities. Clustering is the unsupervised analogue of classification, in that it seeks out discrete classes in data. Dimensional reduction can be similar to regression in that it includes defining a small number of continuous variables that "explain" a larger set of variables. As for classifier, we use the most basic classifier — **k nearest neighbours** — and concentrate on how to process the features to make this simple algorithm work as efficiently as possible.
* Finally, We will chart out characteristic signals (e.g., patterns, separability, relationships between features and targets, relationships between different features, etc.) and fluctuations (e.g., noise levels, data distribution, etc.).Then, we should modify the parameters that affect the creation of the feature vectors and explore different visualizations to help identify a good feature transform.

**Methods**

* There are a plethora of clustering methods available. **K-means** is a simple one that finds clusters using an iterative algorithm that brings clusters slightly closer to their "local centre" with each step. The number of clusters must be determined ahead of time. In general, estimating the number of clusters is difficult, but there are algorithms for doing so.
* Dimensional reduction is a typical unsupervised learning task that involves taking a dataset with a dimension of dh and reducing it to a dimension of dl that is smaller than dh but preserves as much useful information as possible. Since humans are better at processing 2D data and struggle with higher dimensions, visualisation is the most common application.
* The classifier is fixed and we assume to not have any knowledge about the model, therefore, the only thing we can do is to adjust parameters and explore the transforms according to the visualization rather than trial and error with classifier.

**Step 1 is to choose the visualization methods, which is showed below:**

1. ***PCA*** sklearn.decomposition.pca
2. ***LLE*** sklearn.manifold.LocallyLinearEmbedding
3. ***ISOMAP*** sklearn.manifold.Isomap
4. ***tSNE*** sklearn.manifold.TSNE
5. ***UMAP*** umap

Insert picture here

* We can see in the visualization, the 'PCA', 'LLE' ,'Isomap', these tree methods are not clear to show 2D plot for our cognition. Thus, the other two methods 't-SNE' and 'Umap' which can show us the clear visualisation are chosen. Therefore, we will use **'t-SNE'** or **'Umap'** as a tool to explore the visualisation.

**Step 2 is to adjust the parameters by observing the visualisation data.**

* Generally, we will use for - loop to plot the data in 2D and check whether the different types are divided to different areas, which is clustering.

Insert the picture here

**Results**

* **‘decimate’** and **‘feature\_range’ :**

According to the comparison in the Jupyter notebook, we find the **'decimate'**, and **'feature\_range'**, these two parameters do not have huge influence to visualize data. Thus, we keep the ‘feature\_range’ as the **original value** between **(0, 1)** and set ’decimate’ to **2** to make sure the number of features is reduced a little.

Thus, for this part, we set:

Insert code here

* **‘window\_fn’** and **‘feature\_fn’ :**

The other aspect is **'window\_fn'** and **'feature\_fn'.** For  **'window\_fn'**, we find there is no clear improvement in classifying the data. However, in the **‘feature\_fn’** , the huge difference between different features are found, **'dct'** and **'fft'** seem to give the best separation. Class 3 separation is quite good, though Class 2 is close to Class 1 and Class 4 is close to Class 0, there are some outliers of Class 0 as well.

Thus, for this part, we set:

Insert code here

* **‘size’** and **‘step’ :**

The most import part here is the **‘size’** and **‘step’,**  in the visualization, we can see that these two values affects the effect of classification a lot. Besides, it has the hugest uncertainty compares to the other parameters. We think it is a wise decision to put it in the end. For **‘size’**, We can see the visualization that it tends that larger values indeed give more separation, however, thay don't need to be too large as that will lead to incorrect classification. For **‘step’,** it is suitable to set it between 128 to 160, which have the improvement than other values.

Finally, we can set all the parameters into a suitable value:

Insert code here

Then we can use the **K nearest neighbours** classifier to fit the data set (X, y), and get 5 prediction accuracy from 5 test data set.

Insert picture here

Finally, we can get a total score from the test function:

Insert picture here

**Discussion**

* From this case study, it is clear that visualization is useful to reduce a large number of calculations. Besides, the non-linear dimensional reduction method has a better effect than linear dimensional reduction method. We can get the clear information from the non-linear one, such as **t-SNE** and **Umap**. The data are from 4Khz 16 bits mono wave files, which is high dimensional data. Thus, we need to transform them to 2D plot which can give us a clear vision about how the data look like and whether the different classes are divided reasonably.
* In the result, in the figure 5, we can also know test data from challenge\_test\_0, challenge\_test\_2, challenge\_test\_3 can be predicted accurately and the problem is in challenge\_test\_1 and challenge\_test\_4. The accuracy are 0.188 and 0.221 respectively.

The truth is that the model can not predict the green and orange part accurately.

* In conclusion, we consider the visualization is a feasible method to use before deciding which kind of classify to choose and selecting the features. Since the visualization can plot the data in the 2D, for our human, we can judge whether the initial parameter is set suitably at first sight, which means we do not need to work with huge number of data.