Assignment 3: Unsupervised Learning and Dimensionality Reduction

Haoran Yang – ID: hyang412 – Email: hyang412@gatech.edu

The algorithms used in this project:

In this project, we will explore following two clustering algorithms.

1. K-means clustering
2. Expectation Maximization

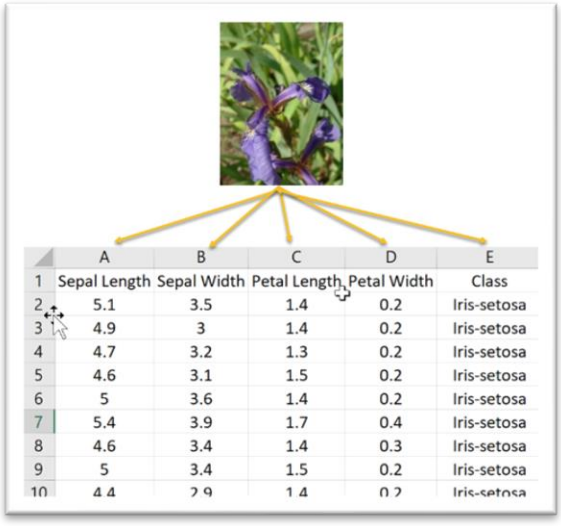
Later, we will apply 4 dimensionality reduction algorithms.

1. PCA
2. ICA
3. Randomized Projections
4. Univariate Feature Selection - (select features according to the k highest Chi-squared scores)

Datasets

To keep the consistency with previous assignments, we will still use Iris data and MNIST handwriting data.

The **Iris data** is perhaps the best-known database to be found in the pattern recognition literature. The data set consists of 50 samples from each of three species of Iris (Iris setosa, Iris virginica and Iris versicolor). Four features were measured from each sample: the length and the width of the sepals and petals, in centimeters. Our task will be to predict the right species by these 4 features.



	A	B	C	D	E
1	Sepal Length	Sepal Width	Petal Length	Petal Width	Class
2	5.1	3.5	1.4	0.2	Iris-setosa
3	4.9	3	1.4	0.2	Iris-setosa
4	4.7	3.2	1.3	0.2	Iris-setosa
5	4.6	3.1	1.5	0.2	Iris-setosa
6	5	3.6	1.4	0.2	Iris-setosa
7	5.4	3.9	1.7	0.4	Iris-setosa
8	4.6	3.4	1.4	0.3	Iris-setosa
9	5	3.4	1.5	0.2	Iris-setosa
10	4.4	2.9	1.4	0.2	Iris-setosa

Figure 1 - Iris Data

The **MNIST data** is a large database of handwritten digits that is commonly used for training various image processing systems, is also widely used for training and testing in the field of machine learning. It was created by "re-mixing" the samples from NIST's original datasets. The black and white images from NIST were normalized to fit into a 28x28 pixel bounding box and anti-aliased, which introduced grayscale levels. The MNIST database contains 70,000 images.

Figure 2 - MNIST handwriting data

Task1: Clustering algorithms

In this section, we will apply two clustering algorithms to two original datasets and conclude some interesting insights.

For K-means clustering, we use Euclidean distances. It should be interesting to compare the clustering results with original labels. Since clustering algorithm just return the n different clusters, there is no 1-1 mapping about the clustering labels to classification labels. To compare them, we will redesign the exact clustering label to get the best results. For example, in Iris data, label the clustering with 1 in which most of its data belong to 1st type of iris plant.

Iris Data

After running a K-means clustering, we easily separate the 150 data points into 3 clusters.

Table 1 - K-means clustering on Iris Data

Confusion Table		Clustering label		
		0	1	2
True label	0	50	0	0
	1	0	48	2
	2	0	14	36

Here all 4 features are be used to cluster and algorithm separates all 150 data points into 3 clusters with size (50, 62, 38). It shows the clustering results based on these four features do have some relationship

with its labels (the type of iris plant). In another word, the three different types of iris plant do show some relevant patterns in their sepal length, sepal width, petal length and petal width. 88% (132 out of 150) different types of iris data can be separated correctly based on this clustering algorithm.

Then we run the EM algorithm and get following results:

Table 2 - EM clustering on Iris Data

Confusion Table		Clustering label		
		0	1	2
True label	0	50	0	0
	1	0	45	5
	2	0	0	50

To some degree, the result is even better than the previous one. Almost all different types of iris plant are separated correctly. Only 5 of 150 are confusing.

What is interesting here is we intuitively evaluate the clustering results by the variable 'type of iris plant', which may not be correct in all circumstances. The aim of clustering is grouping a set of objects in such a way that the objects in the same group are more similar to each other than to those in other groups. During this unsupervised process, there are nothing about the target variable while the final clustering results show high correlations with the target variable in the data. This linking between supervised learning and unsupervised learning provides us more topics to explore in the future.

MNIST handwriting data

After running a K-means clustering, we easily separate the 70,000 data points into 10 clusters. It should be interesting to compare the clustering results with original labels.

Table 3 - K-means clustering on Iris Data

Confusion Table		Clustering label									
		0	1	2	3	4	5	6	7	8	9
True label	0	5506	0	46	292	82	34	248	38	643	14
	1	0	4364	9	9	4	3457	10	7	8	9
	2	74	368	4907	446	246	477	188	72	164	48
	3	37	426	253	4563	83	91	68	48	1406	166
	4	4	116	21	0	2604	241	148	1818	11	1861
	5	68	139	20	2062	228	772	143	394	2169	318
	6	91	295	113	34	464	219	5476	1	181	2
	7	14	248	39	3	707	279	5	3182	3	2813
	8	36	301	51	1578	152	438	65	279	3680	245
	9	40	149	9	93	1718	84	9	1893	55	2908

Here all 784 features are be used to cluster and algorithm separates all 70,000 data points into 10 clusters. Compared with the previous dataset, this clustering result is not so close to the class label. 54% (38K out of 70K) different handing writing numbers are separated correctly based on the clustering algorithm.

When we run the EM algorithm for this dataset, it showed memory error because of the limited computer memory. Thus, in all following parts, we will first sample only half of the data and run the algorithms.

Table 4 - K-means clustering on Iris Data

Confusion Table		Clustering label									
		0	1	2	3	4	5	6	7	8	9
True label	0	2412	76	37	37	2	886	11	2	1	11
	1	48	3261	15	13	0	543	18	0	11	16
	2	1550	237	1178	175	13	212	129	4	6	20
	3	981	710	117	611	10	1086	12	7	2	95
	4	689	341	28	77	166	621	18	119	10	1335
	5	692	239	16	141	16	1876	22	18	21	102
	6	403	391	56	5	69	88	2385	0	12	1
	7	48	306	6	50	117	222	0	663	13	2228
	8	491	646	5	106	19	1754	6	24	5	333
	9	49	578	8	42	203	245	0	156	6	2159

The result is similar to the previous one. 42% (15K out of 35K) different handing writing numbers are separated correctly based on the clustering algorithm.

Task1 Conclusion

1. For iris data, EM algorithm shows a better result according to its original y label. Also, we do not standardize or normalize the features, which may also impact the K-means clustering.

2. For MNIST, K-means algorithm shows a better result. The 784 features are in same range which may improve the k-means algorithm.

3. Compare two datasets, the iris data shows a better result according to its original y label. First, MNIST data has 10 classes while Iris data has only 3. Also, the 4 features in iris data are relatively intuitive and easy to use while MNIST data has 784 features, and it could be really complicated to explore their inside relationships with the class (y label).

Task2: Dimensionality reduction algorithms

In this section, we will apply 4 dimensionality reduction algorithms to two datasets and conclude some interesting insights.

Iris Data

For PCA, 3 new features are created. Their ratio of explained variance are 0.925, 0.053 and 0.017. Clearly, the first new feature explains the most variance in original 4 features.

For ICA, 4 new features are created.

For RAC, 3 new features are created.

For univariate feature selection, we select 3 features according to their univariate Chi-squared scores.

Task3: Reproduce clustering experiments

We apply 4 dimensionality reduction algorithms before clustering and record the % predictions that align with the labels.

Table 5 – Dimensionality Reduction + Clustering [iris data]

% data align with label	Original	PCA	ICA	RCA	Feature Selection
K-means	89.3%	89.3%	81.3%	**98.7%**	**89.3%**
EM	96.7%	95.3%	67.3%	**97.3%**	**98%**

First, dimensionality reduction helps to keep clustering faster and accelerate the process. According to the results, we can see for iris data, both RCA and Feature selection show some improvements.

Task4: Dimensionality reduction application

It is interesting to explore how to connect unsupervised learning with supervised learning. Thus, we apply 4 dimensionality reduction algorithms and feed the new features into the neural network model we built in assignment 1.

Small modifications have been made here to better compare the new results with original results

1. Split the training set and testing set by 50%-50%
2. Set one hidden layer with 100 nodes
3. Limit the max-iterations as 10000

Table 6 - Dimensionality Reduction + Neural Network [iris data]

Neural Network Accuracy	Original	PCA	ICA	RCA	Feature Selection
Testing	97.3%	93.3%	88%	**98.7%**	96%
Training	98.7%	98.7%	98.7%	**98.7%**	98.7%

First, dimensionality accelerate the training of the model. According to the results, it is hard to say that dimensionality reduction will always increase the supervised learning performance. For this iris example, only the RCA brings some improvements in terms of the final accuracies.

Task5: Dimensionality reduction application

After exploring the connection between dimensionality reduction with neural network, we also design an experiment to connect clustering with neural network.

Same modifications have been made as task 4.

Table 7 - Dimensionality Reduction + Neural Network [iris data]

Neural Network Accuracy	Original	K-means	EM
Testing	97.3%	93.3%	97.3%
Training	98.7%	98.7%	98.7%

From the results, there is no obvious improvement from pre-clustering.

Conclusion

- In task 1, clustering algorithms give the results which to some degree align with the class label.
- In task 2, we apply 4 different dimensionality reduction algorithms (filtering) and explain how the new features look likes.
- In task 3, we combine 4 different dimensionality reduction algorithms and clustering algorithms together on iris data.
- In task 4, we apply dimensionality reduction techniques before training supervised learning model
- In task 5, we apply clustering algorithms before training supervised learning.
- Iris data is a good dataset to implement the above 5 tasks. In the future, we can use more complicated dataset (both # of samples and # of features) to further explore.
- In terms of improving supervised learning performance, the impacts of dimensionality reduction and pre clustering highly depend on the dataset and the SL model used. In this sample, neural network is already powerful enough so that we do not observe great improvements.