**학습 알고리즘을 이용한 손 글씨 데이터 분석**

**박상지†**

**Hand-written digit data analysis using learning algorithms**

**Sang Ji Park†**

**ABSTRACT**

Due to the influence of COVID-19, educational applications using mobile devices have increased. It provides a writing environment for efficient learning, such as an increase in educational applications, but it is difficult to recognize when the font shape is relatively irregular, such as handwriting. In the case of handwriting, there is a limit to increasing the recognition rate because it is difficult to prepare various datasets. In order to overcome these limitations, we intend to improve the accuracy of handwriting recognition based on the algorithm learned in Professor Kim Eun-sol's Artificial Intelligence subject. The technologies used were analyzed using a total of five algorithms: 'Logistic Regression', 'Support Vector Classifier', 'K Neighbor Classifier', 'Linear Discriminant Analysis', and 'Multilayer Perceptron'.

Key word : MNIST, Prediction, Deep Learning, Machine Learning

**1. 서론**

코로나19 영향으로 모바일 기기를 활용한 교육용 애플리케이션이 증가하였다. 교육용 애플리케이션의 증가와 같이 효율적인 학습을 위한 쓰기 환경을 제공하고 있지만, 손 글씨와 같이 글씨 모양이 상대적으로 불규칙한 경우 이를 인식하기 어렵다는 점이 존재한다. 손 글씨의 경우 다양한 데이터 셋을 마련하기 어렵기 때문에 인식률을 높이는데 한계가 있다. 이러한 한계를 극복하기 위해 ‘김은솔’ 교수님의 ‘인공지능’ 과목에서 배운 알고리즘을 바탕으로 손 글씨 인식에 대한 정확도를 개선시키고자 한다. 사용한 기술은 ‘Logistic Regression’, ‘Support Vector Classifier’, ‘K Neighbor Classifier’, Linear Discriminant Analyisis’,’ Multilayer Perceptron’ 총 5가지 알고리즘을 사용하여 분석하였다.

**2. 문제 정의**

데이터 생성에 필요한 손 글씨들을 scikit-learn에서 제공하는 ‘dataset.load\_digits()’ 함수를 통해 데이터를 읽어 왔다. 총 데이터의 수는 1797개로 10개의 클래스를 가지고 있다. 특징은 0-16을 가지며 64차원을 나타내고 있다. 총 데이터에서 학습 데이터와 테스트 데이터를 각각 1347, 450로 나누어서 실험을 계획했다.

|  |  |
| --- | --- |
| Class | 10 |
| Sample total | 1797 |
| Training set | 1347 |
| Test set | 450 |
| Dimensionality | 64 |
| Features | (int) 0 -16 |

data set summary table.

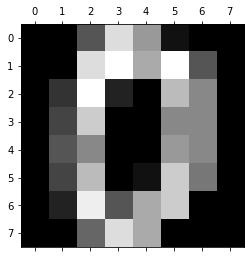


Fig. 0. Handwriting data example.

숫자 데이터는 8x8 픽셀 이미지로 구성되어 있으며, 데이터 세트의 이미지 속성은 각 이미지에 대한 회색 조 값의 8x8 배열을 저장하였다. Fig. 1에서는 테스트 데이터에서 64개의 손 글씨 데이터를 시각화한 것이다. 파랑 색 글씨는 데이터의 ‘target’ 속성이며 실제 이미지가 나타내는 숫자를 의미하며 후에 모델 학습에 대한 예측 값이 target 값과 비교하여 올바르면 파랑색을 오답이면 빨강색으로 출력하게 설정하였다.

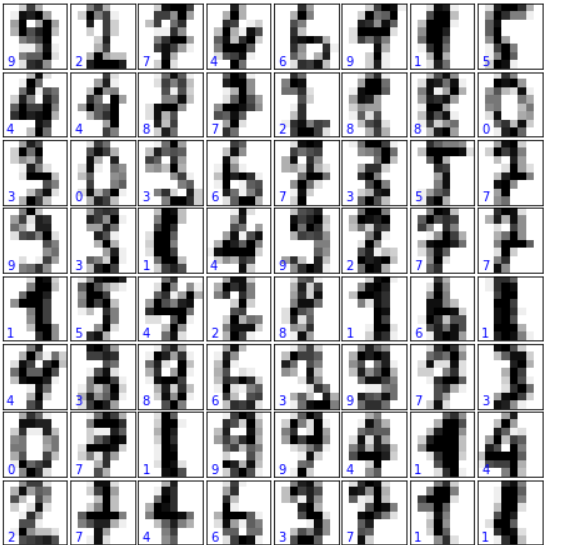


Fig. 1. Handwriting data visualization

**3. 실험 및 결과**

Logistic Regression(LR)은 회귀를 사용하여 데이터가 어떠한 범주에 속할 확률을 0에서 1사이의 값으로 예측하고 그 확률에 따라 가능성이 더 높은 범주에 속하는 것으로 분류해주는 지도 학습 알고리즘이다. ‘sklearn’에 내장된 ‘LogisticRegression’ 모듈을 사용하였다. LR 모듈로 Training set을 학습하기 전에 ‘sklearn’에 내장되어 있는 ‘StandardScaler’로 Training set의 특징들을 평균이 0이고 분산이 1로 표준화를 시켜주었다. 표준화된 데이터를 가지고 Logistic Regression 모델로 학습을 시켰다. Fig. 2에서는 실제 값과 예측 값에 대한 혼동 행렬(confusion matrix)을 나타낸 것이다.

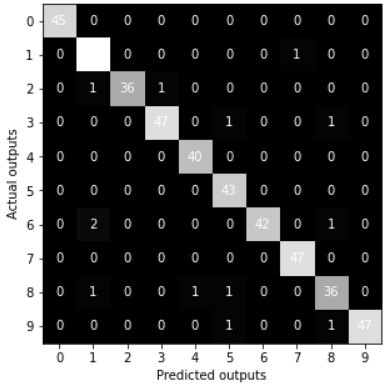


Fig. 2. LR confusion matrix.

LR 모델로 손 글씨 데이터를 분석한 96% 정확도를 보였다.

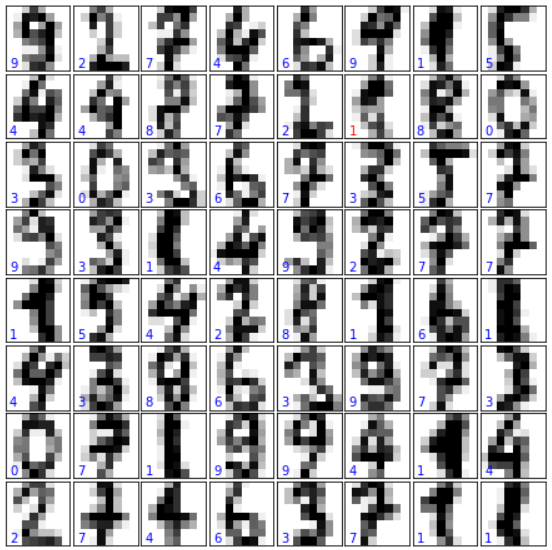


Fig. 3. LR Handwriting data visualization

Linear Discriminant Analys(LDA)은 데이터 분포를 학습해 결정경계를 만들어 데이터를 분류하는 모델이다. LDA는 데이터를 특정 축에 사영을 한 후에 두 범주를 잘 구분할 수 있는 직선을 찾는 것을 목표로 한다.

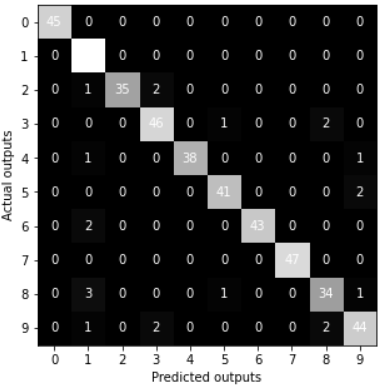


Fig. 4. LDA confusion matrix

LDA 모델로 손 글씨 데이터를 분석한 결과 95%의 정확도를 보였다.

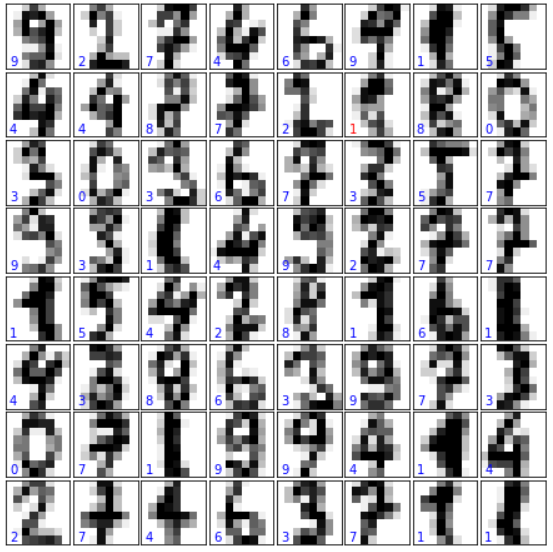


Fig. 5. LDA Handwriting data visualization

K Neighbor Nearest(KNN)은 새로운 데이터를 입력 받았을 때 입력의 특징 공간 내 k개의 가장 가까운 훈련 데이터로 구성된다. 이 때, 데이터 간 거리를 측정할 때는 유클리드 거리계산법을 사용하였다. KNN은 거리 기반 모델이기에 모델 구현 시 표준화를 해야 한다. 클래스간의 나타내는 범위의 척도가 다르기 때문에 해당 범위를 조정하지 않게 되면, 범위가 큰 값은 거리 함수에 큰 영향을 미치게 된다. 데이터간의 분류에서 정도에 따라 잘못된 분류를 방지하기 위해서 StandardScaler 함수를 이용해서 정규화 작업을 하였다.

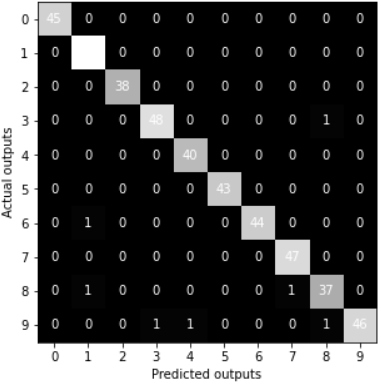


Fig. 6. KNN confusion matrix

KNN 모델로 손 글씨 데이터를 분석한 결과 97%의 정확도를 보였다.

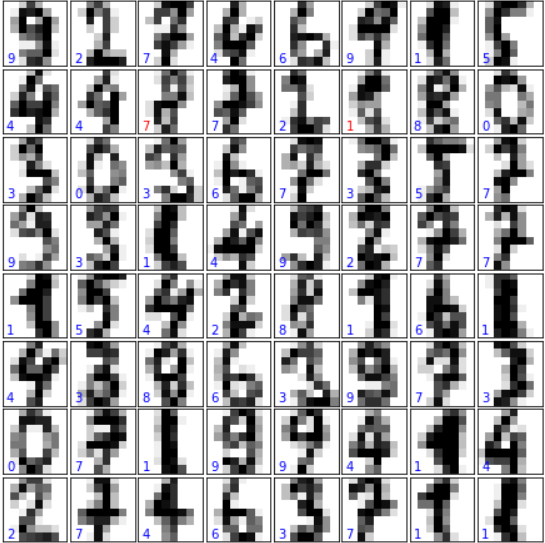


Fig. 7. KNN Handwriting data visualization

Support Vector Classifier(SVC)은 분류를 위한 경계를 명확하게 만들기 위해 고차원으로 변환해 학습하는 모델이다. 큰 차원으로 변환되는 복잡한 분류 경계도 명확히 할 수 있으며 다른 모델들보다 과적합되는 경향이 적다.

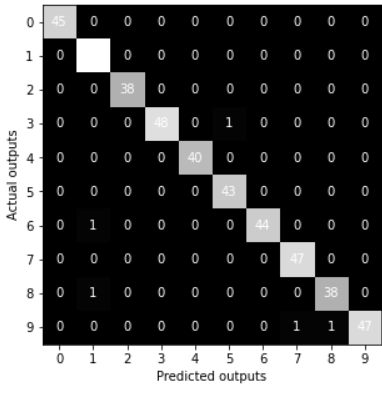


Fig. 8. SVC confusion matrix

SVC 모델은 손 글씨 데이터를 분석한 결과 98%의 정확도를 보였다.



Fig. 9. SVC Handwriting data visualization

Multilayer Perceptron(MLP)는 여러 개의 퍼셉트론 뉴런을 여러 층으로 쌓은 다층신경망 구조로, 각 층 안에서 뉴런간 연결이 없으며 인접한 두 층의 뉴런 간에는 완전이 연결된 구조이다. 퍼셉트론과 마찬가지로 값을 넣고 결과값이 나오면, 퍼셉트론의 결과 값과 실제 결과값을 비교를 한 뒤 오류를 전파을 하는 원리는 똑같다. 은닉층을 많이 쌓고, 퍼셉트론들을 많이 생성할수록 학습하는데 오랜 시간이 걸리며, 최적화된 은닉층의 개수와 퍼셉트론 개수를 찾기가 어렵다. 손 글씨 분석에서는 은닉 층(hidden Layers)의 크기를 (10,30,10)으로 지정하여 분석하였고, 활성화 함수(activate function)는 relu로 설정하였다.

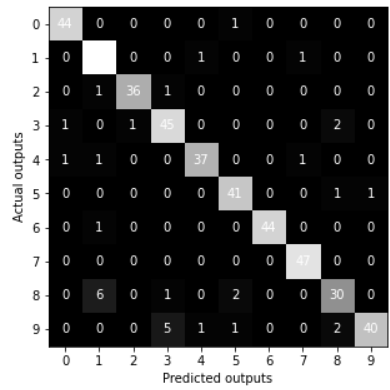


Fig. 10. MLP confusion matrix

MLP 모델은 손 글씨 데이터를 분석한 결과 92%의 정확도를 보였다.

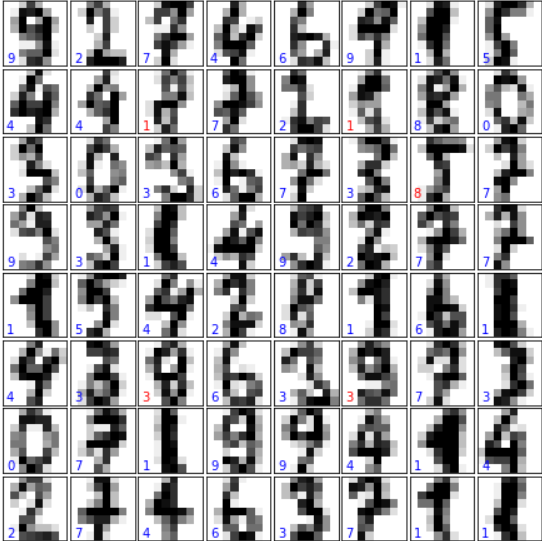


Fig. 11. MLP Handwriting data visualization

**4. 결론**

Table 1. test time and accuracy of algorithms performance

|  |  |  |
| --- | --- | --- |
| Model | Time [sec] | Accuracy [%] |
| LR | 0.32 | 97 |
| LDA | 0.03 | 98 |
| NN | 0.48 | 95 |
| KNN | 0.002 | 98 |
| SVC | 0.14 | 85 |

손 글씨 데이터를 분석하는데 있어서 가장 높은 예측 정확도를 가진 알고리즘은 KNN과 LDA이었고, 테스트 시간마저 두 알고리즘이 좋은 성능을 보여주었다.

향후 손 글씨 데이터를 분석하는데 있어서 더 높은 예측 정확도와 처리 속도를 개선을 위해 대량의 손 글씨 데이터을 지속적으로 확보하고, 개별적인 알고리즘을 조합하여 다중 알고리즘으로 손 글씨 데이터를 분석을 계획하고 있다.

**REFERENCE**

[1]<http://ce.sharif.ir/courses/85-86/2/ce725/resources/root/LECTURES/SVM.pdf>

[2]<https://machfam.com/16450>

[3]<https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html>

[4]<https://scikit-learn.org/stable/modules/generated/sklearn.discriminant_analysis.LinearDiscriminantAnalysis.html>

[5]<https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html>

[6]<https://scikit-learn.org/stable/modules/generated/sklearn.datasets.load_digits.html>

[7]<https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.KNeighborsClassifier.html>

[8]<https://scikit-learn.org/stable/modules/generated/sklearn.neural_network.MLPClassifier.html>

[9]<https://coderzcolumn.com/tutorials/machine-learning/scikit-learn-sklearn-linear-dimensionality-reduction-pca>

[10]<https://scipy-lectures.org/packages/scikit-learn/auto_examples/plot_digits_simple_classif.html>

[11]<https://docs.w3cub.com/scikit_learn/modules/generated/sklearn.datasets.load_digits>