ML_Lab Final Project Presentation

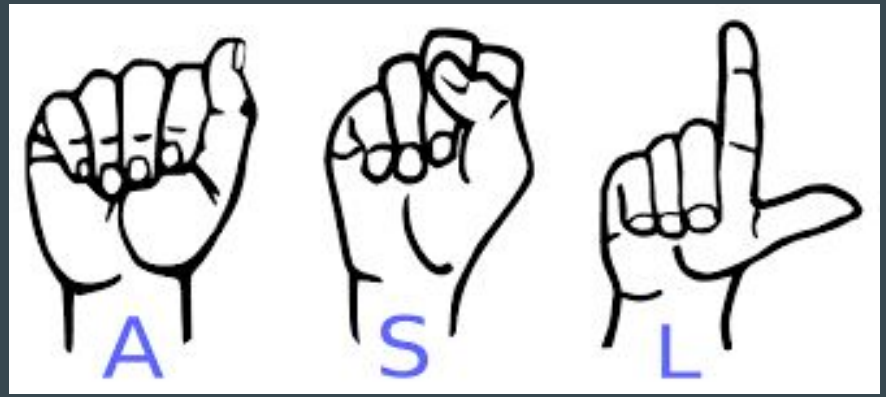
By Anthony Allen, Parth Patel, Caijun Qin

OUTLINE

- Project Description
- Approach
- Conclusion

Project Description

- Use Machine Learning to recognize letters of ASL alphabet.
- Image dataset of letters A - I generated from various sources
- Image size: 100x100x3
- Goal of 90% accuracy score



Approach

Pre-Processing

- ➔ Grayscale Conversion
- ➔ Image Vectorization
- ➔ Pixel-Normalization

Train Model

- ➔ K-Nearest Neighbors Classifier
- ➔ Single neighbor
- ➔ Uniform Weight for Minkowski Metric

Evaluate Model

- ➔ Score Function associated with KNN Model.
- ➔ Based on accuracy as score metric.

Comparison of Models

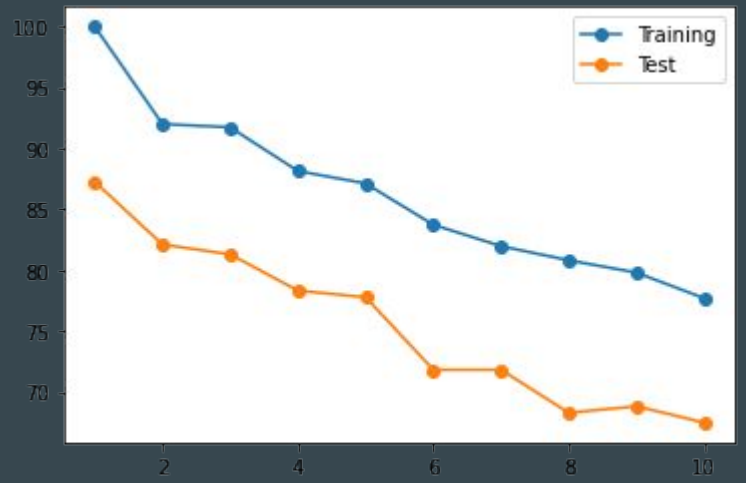
- Hyperparameter Tuning (pixel normalized data, no feature extraction)
  - LDA (6 Components)
  - kNN (Uniform Weights, Minkowski Metric, P=2, n_neighbors = 1)
    - Accuracy vs n_neighbors
  - Decision Tree (max_depth = 12, min_samples_leaf=4)
    - Overfit training data
  - Random Forest (N_bins = 20)
  - SVM(C = 20, gamma = 1000)
    - Second best choice
  - MLP(hidden_layer_sizes = 72)

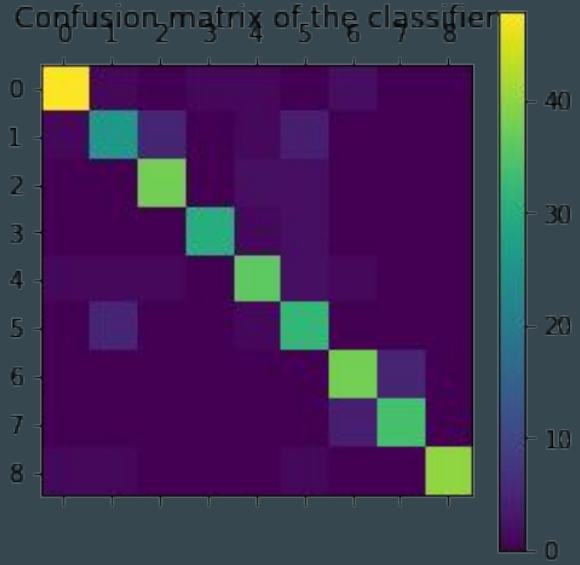
Comparison of Models

TABLE II
BEST ACCURACY SCORES FOR EACH MODEL

Classifier	Parameters	Training Accuracy	Testing Accuracy	Mean Average Error
LDA	N_components = 6	93.22	60.43	2.08
kNN	N_neighbors = 3	96.80	90.50	0.35
Decision Tree	depth=12 leaf = 3	85.67	53.46	1.52
Random forest	N_Bins = 50	100	79.40	1.52
SVM	C = 20 Gamma = 1	96.45	83.97	0.45
MLP	Batch = 72	73.21	64.58	2.85

Selecting the Best Model





Less number of neighbors led to increased accuracy.

Heatmap of the KNN model's confusion matrix.

Improving kNN

- Create pipeline (grayscale, image vectorization, pixel normalization, kNN, accuracy_score)
- Tested with HOG feature extraction
  - kNN typically works poorly with large datasets
- Tested with PCA feature extraction
- Pre-processing techniques also analysed
  - kNN sensitive to noisy data
  - Robust, edge detection, smoothing, masking

Improving kNN

TABLE III
ACCURACY SCORE OF PIPELINE NORMALIZATION VS FEATURE EXTRACTION

	NONE	HOG	PCA
Pixel Normalization	90.50	57.99	53.12
Min-Max	84.23	52.85	51.28
Standard scalar	86.44	55.55	52.85
Robust Scalar	86.17	88.08	51.49
Smoothing	88.90	88.89	58.26
Sobel	65.32	70.73	38.48
Masking	65.32	62.30	13.45

* scores represent percentages

Conclusions

- kNN
- No feature extraction
- Pixel normalized data
- Preprocessing made a significant impact on accuracy scores