ML : Dog Breed And Center Point Detection

Omar El khyari, Ziad Chentouf, Omar Zakariya
Project 1 of Machine Learning, EPFL

I. Introduction

Our project assessed Linear Regression, Logistic Regression, and K-Nearest Neighbors (KNN) for two tasks: determining dog center points and breed classification in images. We compared method performances across varied validation splits to gauge their effectiveness.

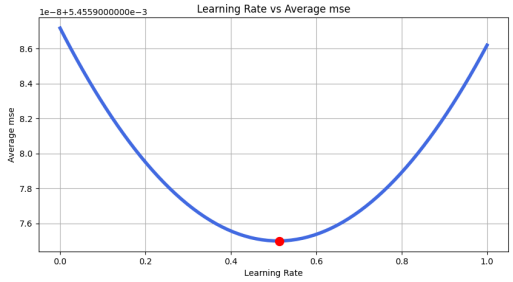
II. Linear Regression for Center Locating

A. Data Preprocessing

Optimizing our Linear Regression (lmda = 0) model involved several experiments to understand the influence of different data pre-processing steps on model performance.

Preprocessing Strategy	Train Loss	Test Loss
No normalization or bias term	0.04933	0.05104
Normalization only	0.26204	0.26102
Bias term only	0.00543	0.00463
Normalization and bias term	0.00543	0.00463

B. Hyperparameter Tuning for Lambda : 5-fold cross-validation



C. Results

We found the optimal lambda at 0.51.

- Train loss: 0.00543 – Test loss: 0.00463

D. Conclusion

The optimal lambda and the inclusion of a bias term were instrumental while normalization had less impact, confirming the effectiveness of our linear regression approach for locating the dog's center in images.

III. Logistic Regression for Dog Breed Identification

A. Data Preprocessing

We assessed the impact of preprocessing strategies by setting learning rate to 1e-5 and iterations to 200. The results are summarized below:

Preprocessing Strategy	Train Accuracy	Test Accuracy
No normalization or bias term	66.508%	62.080%
Normalization only	86.521%	85.321%
Bias term only	84.888%	84.098%
Normalization and bias term	87.577%	86.544%

B. Finding the Best Learning Rate and Iterations : 5-fold cross-validation

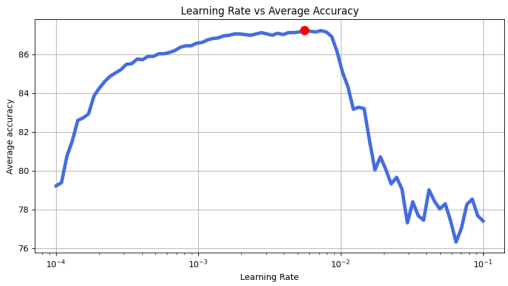


Fig. 1: Optimal learning rate.

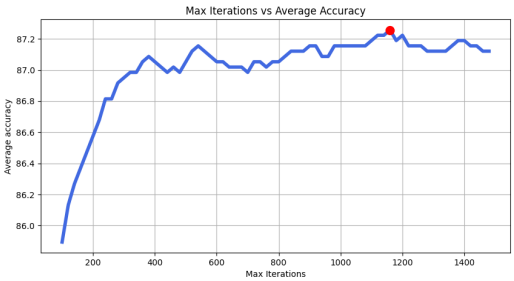


Fig. 2: Optimal number of iterations.

C. Results

Optimal parameters found were a learning rate of 0.00558 and 1160 iterations, achieving high accuracy and F1-scores:

- Train set: accuracy = 87.8%, F1-score = 0.8736
- Test set: accuracy = 86.8%, F1-score = 0.8575

D. Conclusion

By fine-tuning lr and max-iters via 5-fold cross-validation and exploring preprocessing options like normalization and bias inclusion, we significantly enhanced the model's performance and generalizability.

IV. K-NN for Dog Breed Identification and Center Locating

A. Data Preprocessing

Data normalization is crucial, especially since KNN is sensitive to the range of data points. This step is important to ensure that outliers do not disproportionately influence the results.

B. Optimizing

1) Regression Task: Center Locating: 5-fold Cross-validation helped us discover that a larger 'k' value tends to yield better results for regression.

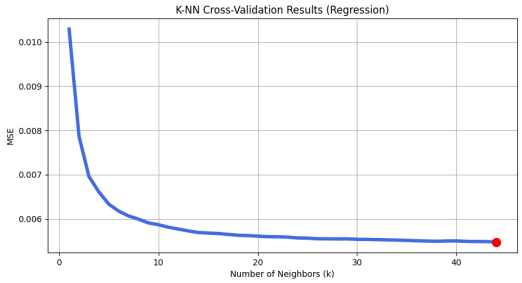


Fig. 3: K-NN 5-fold Cross-Validation for Regression.

For 'k' equal to 400, the results were promising:

- Train loss: 0.00541
- Test loss: 0.00463

2) Classification Task: Breed Identifying: Similarly, for classification, 5-fold cross-validation was employed to find the optimal k = 7.

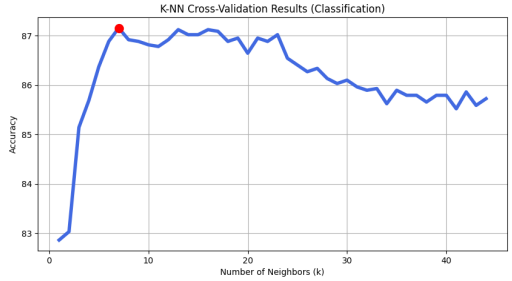


Fig. 4: K-NN 5-fold Cross-Validation for Classification.

leading to high accuracy and F1-scores:

- Train set: accuracy = 89.210% - F1-score = 0.888286
- Test set: accuracy = 87.768% - F1-score = 0.863541

C. Conclusion

Through hyperparameter tuning using 5-fold cross-validation, notably the selection of 'k', and normalization, our model achieved high accuracy.

V. Validation Set

A. Data Preparation and Validation Set Selection

The choice of a validation set partition is crucial for a model's ability to generalize to unseen data. We explored different split ratios to identify the most effective validation proportion for our models.

B. Experimental Setup using optimal parameters

- For classification: KNN (K=7), LR=0.0055825, max-iter=1160.
- For regression: KNN (K=400), lambda=0.51.

C. Results and Analysis

The following figures showcase the model performances across different validation splits.

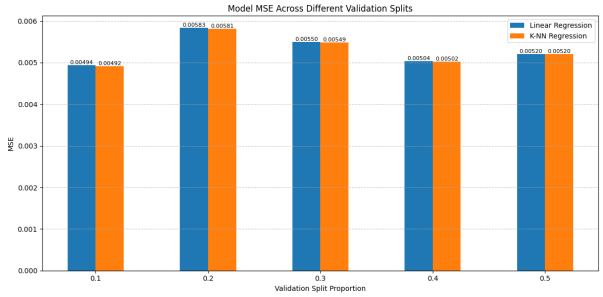


Fig. 5: Performance for Regression

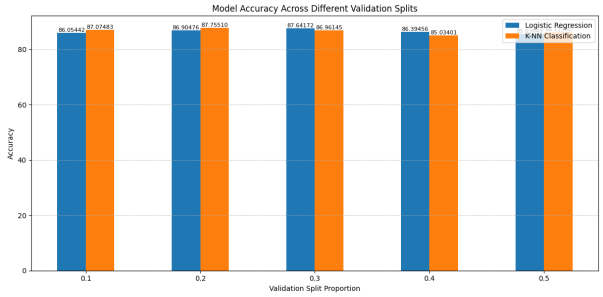


Fig. 6: Performance for Classification

D. Performance Evaluation on Test Data

For the final evaluation, we applied the models to test data using the best validation splits identified: a 10% split for regression tasks and a 20% split for classification tasks even though the model was less sensitive to split proportions.

- For Linear Regression (lambda=0.51): Train loss=0.0053, Test loss=0.0061.
- For K-NN Regression (K=400): Train loss=0.0053, Test loss=0.0055.
- For Logistic Regression (LR=0.0055, max iters=1160): Train accuracy=87.8%, Test accuracy=87.5
- For K-NN Classification (K=7): Train accuracy=88.9%, Test accuracy=87.9%