Machine Learning (Problem set 4)

Ali Alipour, *University of Tehran*

This report presents a series of machine learning experiments focusing on the classification of datasets using different algorithms and techniques. The tasks involve splitting data into training and testing sets, applying models such as linear regression and support vector machines (SVM), and evaluating their performance through accuracy metrics and confusion matrices. The report first explores the use of a linear model to classify features, followed by an analysis of the impact of feature selection on model accuracy. Next, we delve into the application of SVM with different kernel functions, including the Radial Basis Function (RBF), linear, and polynomial kernels. Each kernel is evaluated based on its ability to separate non-linear data, and regularization techniques are employed to control overfitting and improve model generalization. Lastly, various multi-class classification strategies, such as one-vs-one and one-vs-rest, are compared for different kernel methods. The performance of these strategies is assessed using confusion matrices and accuracy scores, providing insights into their effectiveness in handling complex datasets. Throughout the report, the experiments highlight the trade-offs between model complexity, accuracy, and computational efficiency.

Question 1

1) We begin by loading the dataset and splitting it into training and test sets. The linear model is applied, and accuracy metrics for the test and training sets are reported.

2) The confusion matrix is also calculated, showing the accuracy of the model on the selected features. The results indicate that the selected features allow for good separation between the classes.

3) We proceed by analyzing the features ‘Petal Length‘ and ‘Petal Width‘. Although the accuracy drops slightly, we still achieve high accuracy for the class ‘Setosa‘, which has a large distance from other classes.

Question 2

We compare the performance of various models (Linear, RBF, Polynomial) using one-vs-one and one-vs-rest strategies. Confusion matrices and accuracy scores are reported for each, with the polynomial kernel showing better performance in multi-class classification.

1. First, we define the necessary functions and libraries. We generate random data points within a specified radius and display the corresponding images.

2. Next, we repeat the process for a second scenario, generating and displaying the random data points again.

3. Using the generated data, we organize them into a DataFrame and assign labels to them.

4. We then write a manual code to split the data into training and testing sets.

5. Using the Logistic Regression class, we manually implement a logistic regression algorithm to classify the data.

6. Afterward, we increase the features using a mapping function to enhance the classification process.

7. Finally, we implement a classification algorithm using radial data generated from uniform and normal distributions and report the results.

1. First, we load the dataset. We define both the Gaussian kernel and the estimated kernel function.

2. We extract the "duration" column from the dataset and represent it as a list. We proceed to plot the results of the kernel density estimations for different values of "duration."

3. We use the `kernelDensity` function to compute the kernel density and plot the results.

4. Lastly, we perform Parzen window estimation for a subset of 250 data points and visualize the results.

References

[Alipour Fraydani, 2024] Alipour Fraydani, A. (2024). Homework on Machine Learning problem set 4, University of Tehran. *Unpublished Manuscript*, Department of Electrical Engineering, University of Tehran.