Ad Click Prediction

IE506:Machine Learning: Principles and Technique

Boddu Siva

Venkata Sri Teja

22N0459

Jan’23-May’23

1 Introduction

In this project, we explore the task of enhancing the performance of multi-class classification models using hyperparameter optimization and regularization techniques. We focus on several popular algorithms, including Linear Regression, Support Vector Machines (SVM), k-Nearest Neighbors (KNN), Decision Trees, and Random Forests. Our goal is to achieve improved predictive accuracy by fine-tuning hyperparameters and implementing appropriate regularization methods.

2 Data Description

This data was collected from text ads found on twelve websites that deal with various farm animal related topics. Information from the ad creative and the ad landing page is included. The binary labels are based on whether or not the content owner approves of the ad. For each ad, we include the words on the ad creative and the words from the landing page. Each word from the creative is given a prefix of 'ad-'. Title and header HTML markups are noted in a similar way in the text of the landing page. We have already performed stemming and stop word removal. Each ad is on a single line. The first word in the line is the label of the instance. It is 1 for accepted ads and -1 for rejected ads

Dataset Link: https://archive.ics.uci.edu/dataset/218/farm+ads

3 Data Preprocessing

Our dataset is characterized by high-dimensional and sparse features in a label-feature format. To handle this data structure effectively, we employed techniques to manage the sparsity and dimensionality. Additionally, we transformed the data into a suitable format for training and evaluation.

4 Hyperparameter Optimization

Hyperparameter optimization is crucial for fine-tuning the performance of machine learning models. We adopted different strategies for each algorithm:

- For Linear Regression, SVM (with RBF kernel), KNN, Decision Trees, and Random Forests, we focused on finding the optimal hyperparameters through an exhaustive search. This involved evaluating the models’ performance over a range of hyperparameter values.

5 Regularization Strategies

Regularization plays a pivotal role in controlling model complexity and preventing overfitting. In our project, we explored both L1 and L2 regularization techniques and their implications:

- **L1 Regularization**: We applied L1 regularization to Logistic Regression and SVM models. L1 regularization encourages sparsity in the model’s coefficients, enabling feature selection and improving interpretability. This is particularly useful when dealing with high-dimensional and sparse data.

- **L2 Regularization**: We employed L2 regularization across all algorithms, including Linear Regression, SVM, KNN, Decision Trees, and Random Forests. L2 regularization mitigates overfitting by penalizing large coefficient values, leading to smoother models. It is effective in preventing overfitting when the dataset is low-dimensional and dense.

- **Elastic Net Regularization**: In practice, a combination of L1 and L2 regularization, known as elastic net regularization, can provide a balanced trade-off between sparsity and smoothness. Depending on the dataset's characteristics, practitioners can consider this approach to achieve better model performance.

6 Hyperparameter Ranges

For each algorithm, we defined specific hyperparameter ranges for optimization:

- Logistic Regression (L2 regularization): We explored a range of C values using the np.logspace(-3, 3, num=4) function. This range covered the trade-off between model complexity and generalization performance.

- SVM (RBF kernel): We explored the γ parameter over the range of 10^{-9} to 10^{3}, with 13 logarithmically spaced values.

- KNN: The number of neighbors was chosen from 1 to $\sqrt{N}$ with a step size of 2, ensuring consideration of odd values for a wide range of neighborhood sizes.

- Decision Trees: We considered a range of minimum weight fraction values for leaf nodes: 0.05, 0.1, 0.15, and 0.2.

- Random Forests: The number of estimators (trees) was explored across four values: 10, 100, 500, and 1000.

7 Evaluation Metrics

To assess model performance, we employed a comprehensive set of evaluation metrics including Accuracy, Precision, Recall, Specificity, and Sensitivity. We evaluated the models on both training and testing datasets to ensure a comprehensive understanding of their generalization capabilities.

8 Conclusion

In this project, we successfully enhanced multi-class classification models through hyperparameter optimization and regularization techniques. By optimizing hyperparameters and applying appropriate regularization strategies, we achieved improved model performance across various algorithms. The choice between L1 and L2 regularization, or a combination of both, depended on the dataset's characteristics and the specific problem at hand. This project highlights the importance of careful consideration of hyperparameters and regularization methods in achieving accurate and reliable machine learning models.

9 Future Work

In the future, further investigations could include exploring more advanced regularization techniques, such as dropout in neural networks, and experimenting with different evaluation metrics or ensemble techniques to achieve even better model performance. Additionally, investigating the impact of feature engineering and data augmentation could contribute to further improving the models' generalization capabilities.