Refining Algorithms

Model hyperparameters are parameters that cannot be derived from the data and therefore must be specified before utilising the model. Default parameters are initially used at the beginning of the machine learning process and are adjusted after model selection. The aim of optimisation is to find a set of parameters which minimise a predefined loss function; exponential loss in adaboost and log loss in logistic regression [1].

Figure 1: Formularisation of hyperparameter search. Where represents the hyperparameters, X (te) represents the test data, represents the training data [1].

Figure one shows the mathematical representation of hyperparameter search. There are many algorithms which automate this process. The most commonly used are grid search and Bayesian search. These algorithms are preferential to the alternative of manually adjusting the parameters, as research shows this results in models with greater predictive performance [2]. Bayesian search was not used due to the time constraints. The sklearn library at the time of conducting the project, did not have a dedicated class to utilise the algorithm and therefore would have required significant time to code. Hyperparameters were evaluated using a tenfold cross validation (cv). Initially kbest was included in the grid search but then removed. Including kbest resulted in poor predictive performance on the test data for the adaboost models, accuracy decreased from 99.07% to 50%. This was the result of the adaboost models overfitting the training data.

For the final three pipelines, the parameters which were tuned are as follows: number of estimators, learning rate, ‘C’ and solver. The number of estimators (N) and learning rate (L) are a parameter of adaboost. There is a trade-off between N and L, the optimal number of estimators required increases as L decreases [3]. The number of estimators is the number of models that are iteratively trained: the values searched were 10,50,100,150. The learning rate is the contribution each model has on the weights of the algorithm. The values 0.001,0.01,0.1 were searched, as past studies suggest values below 0.1 are optimal [3]. The lowest L values should have the greatest cross validation score as a result of reduced overfitting however, the opposite was true. The highest L achieved the best cv score. The parameters Solver and ‘C’ are used to optimise logistic regression. The inverse of regularisation strength is referred to as ‘C’. The lower the ‘C’ the greater the regularisation strength. This reduces the variance of the model and therefore should reduce overfitting however, the highest C values resulted in the greatest cv score [4].

Reference

[1]M. Claesen and B. De Moor, “Hyperparameter Search in Machine Learning,” in *MIC 2015: The XI Metaheuristics International Conference in Agadir, Morocco*, 2015.

[2]J. Bergstra and Y. Bengio, “Random Search for Hyper-Parameter Optimization Yoshua Bengio,” *Journal of Machine Learning Research*, vol. 13, pp. 281–305, Feb. 2012.

‌

[3]S. Kim, M. Son, K. Kwon, and D. Lee, “Electricity Demand Forecasting Algorithm Development Based on the Gradient Boosting Machine by Selecting the Optimal Combination of Weather Data and Stations,” *The transactions of The Korean Institute of Electrical Engineers*, vol. 68, no. 7, pp. 834–843, Jul. 2019.

‌[4]G. James, D. Witten, T. Hastie, and R. Tibshirani, *An introduction to statistical learning : with applications in R*. Springer, 2013.

‌

‌