Design and Implementation of Hyper Parameter Tuning Working Principle to Predict Brain Tumor

[1] Y.Tezaaw, 2 Dr.K.Vijaya Lakshmi
[1] Research Scholar, [2]Associate Professor
[1, 2] Dept.of Computer Science, S.V.University, Tirupati, A.P,India

Abstract: *A gene is a sequence of instructions that instructs a cell to make proteins or other molecules based on the information in DNA. Translation involves converting DNA into messenger RNA (mRNA) and then converting that into proteins. The order of genetic changes occurring in a tissue or single cell under specific circumstances is evaluated using gene expression analysis. Cancer refers to a group of disorders in which the human body develops malignant cells as a result of genetic mutation. As they mature, these cells are divided randomly and spread over the organs and, in many instances, can cause death. Primary brain tumors develop from the brain's own cells, whereas secondary brain tumors develop from cancerous cells that have travelled to the brain from elsewhere[7]. Studies have revealed that brain tumors are incredibly diverse, making categorization, segmentation, prediction and diagnosis are extremely difficult. DNA microarray technology has also substantially altered about the understanding of what causes cancer in recent years. Small sample sizes and a wide range of gene expression levels in cancer microarray data can result in the "curse of dimensionality," which makes it challenging to categorize the data. ML methods are used by the bioinformatics community to categorize microarray data in various ways. The majority of the studies using microarray data sets for cancer classification that have been examined focus on accuracy in cancer classification without disclosing relevant biological information about the cancer classification approach. Microarray data sets can be interpreted biologically as well as classified accurately by different models in this implementation.*

Keywords: *Brain Tumor, Microarray data sets, Hyper Parameter Tuning Working Principle, Lazy predict, Untuned and Tuned SVC*

I. Implemetation of Hyper Parameter Tuning

Working Principle

Let us consider the feature variable as X and target variable as Y with an unidentified joint distribution as D(X,Y) in which the sample dataset is segregated as S with m observations. The ML model can acquire data using functional relationship among X and Y by generating a prediction model as $\hat{F}(X,\ \theta)$ that can be controlled through n-dimensional hyperparameter configuration $\theta = (\theta_1, \theta_2, \dots, \theta_n)$ from the hyperparamater search spacing $\ominus = (\ominus_1, \ominus_2, \dots, \ominus_n)$. The prediction performance has been calculated through point-wise relationship among prediction function $\hat{F}(X,\ \theta)$ and true label Y. The loss function is defined as $L\left(Y, \hat{F}(X,\ \theta)\right)$ and usually the expected risk of the involved algorithm are measured based on recent data and also sampled from $D: R(\theta) = E\left(L\left(Y, \hat{F}(X,\ \theta)\right)\middle| D\right)$. The particular data distributions are provided by mapping encoder with convinced learning model and for particular performance measurement whereas the numerical defined to all hyper parameter configuration is θ.

The providence of k various datasets with data distributions is expressed as $D_1, D_2, \dots, D_k$. This is illustrated for k hyperparameter risk mapping is shown in equation 1.

$R^{(i)}(\theta) := E\left(L\left(Y, \hat{F}(X,\ \theta)\ \right)\middle| D_i\right)$ i=1,2,,k
(1)

II. Setting of Optimal Hyperparameter configuration

Initially, let consider the best hyperparameter configuration for i dataset is expressed in equation 2.

$$\theta^{(i)*} := \underset{\theta\epsilon\ominus}{Arg}\min R^{(i)}(\theta)$$
(2)

However, the general setup have been presumed to endeavor sufficiently over various dissimilar datasets and normally determined using software packages with frequent ad hoc or heuristic way. Therefore, the optimal hyperparameter configuration is obtained with respect to comprehensive empirical experiments with respect to k dissimilar benchmark datasets is expressed in equation 3.

$$\theta^{*} := \underset{\theta\epsilon\ominus}{Arg}\min g\left(R^{(1)}(\theta), R^{(2)}(\theta) \ldots, R^{(k)}(\theta)\right)$$
(3)

Where,

g = summarized function of specified

$R^{(i)}(\theta)$ = Expected risk for mapping hyperparameter

Moreover, the $R^{(i)}(\theta)$ estimation has been potentially scaled suitably before making it highly proportional among datasets that assist to scale any $R^{(i)}(\theta)$ to [0,1] by eliminating the outcome with reference to dummy predictor and divide it by absolute difference among those best potential predictor or calculated through Z-score. The suitable scaling is majorly based on measuring performance.

III. Algorithm for Randomized Search

Step 1 – Initializing the origin point as $X_0 \subset S$, algorithm parameters as Θ_0, and the iteration index as k = 0.

Step 2 – Generating the candidate point's collections as $V_{k+1} \subset S$ with respect to the certain generator as well as the relative sampling distribution.

Step 3 – Considering the candidate points $V_{k+1,}$ X_{k+1} get updated related to prior iterations as well as algorithmic parameters and possibly upgrade the settings of the algorithm as Θ_{k+1}.

Step 4 - If a stopping criterion get mapped then stop else proceed with incremental in k and return to Step 2.

Step 5 – The modified technique in Step 3 and the generator in Step 2 that creates candidate points both serve as the foundation for the random search principle.

Step 6 - Returns

The complete implementation is done using Jupyter notebook and are executed on Google Colaboratory and the model is stored in Google Drive.

IV. Results and Discussion

In this work, Google Colab is used with Jupiter IDE which assist to share and create document that can be narrated with text, livecode and visualizations. The student personal database as dataset is collected and split it into 60% train dataset and 40% test dataset. The tunability hyperparameter used other tools like Scipy, Seaborn and Pandas. From the top most best performed ML model is considered with untuned ML model. Improvement of model accuracy is based on the parameter of regression is enhanced through proposed Randomized Search CV hyperparameter tuning. Lazy predict is best python libraries which assist for semi-automate the ML based tasks. Lazy predict help in building several and different basic ML models with certain code and assist in understanding which models may execute better accuracy without any parameter tuning. In this research lazy classifier is used for solving regression based dataset in predicting the brain cancer classification accurately to provide earlier treatment. From the lazy classifier analysis, SVC is consider to be top most model which is evaluated through confusion matrix metrics is shown in Figure 1.

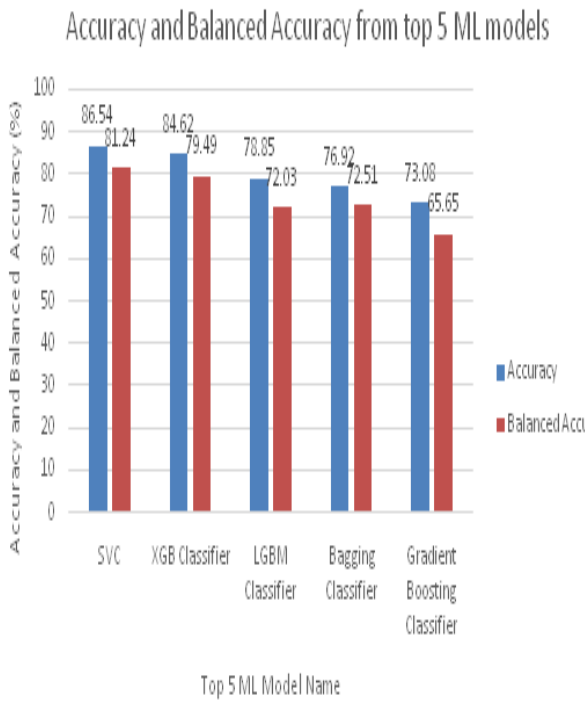

Figure1: Accuracy and Balanced accuracy for top 5 ML models

Figure 1 illustrate the accuracy and balanced accuracy for top five high accuracy classifier model from lazy predict which execute using the in-built library. According to the data, the SVC has performed high accuracy and balanced accuracy as 86.54% and 81.24% than other classifier ML model like XGB classifier, LGBM classifier, Bagging classifier and Gradient Boosting classifier.

Figure 2 illustrate the time consumed for top five high accuracy classifier model in which time consumed for the model training and execution time is 0.005 Sec. The SVC time consumption is very lesser than other ML classifier model. This assist in producing high accuracy with less time consumption.

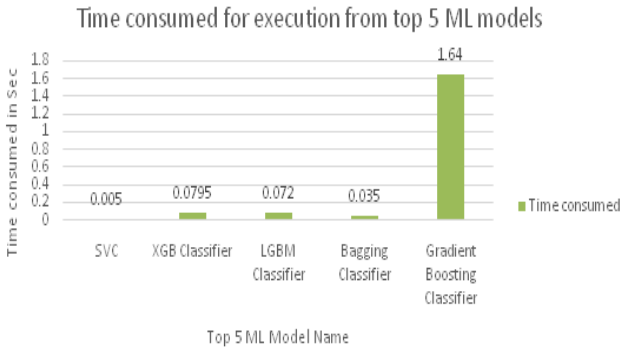

Figure 2: Time consumed for top 5 ML models

All the above-mentioned classification techniques are implemented using the above specified tools. In order to find out the efficient untuned regression technique for prediction. SVC is identified as best untuned ML model and the parameters of SVC is represented with range of parameters and Randomized search CV parameter distribution is listed and iterated with hyperparameter tuning. The tuned SVC is evaluated with untuned SVC through confusion matrix metrics like accuracy, micro and weighted precision, recall and F1-Score shown in Figure 3 and Figure 4.

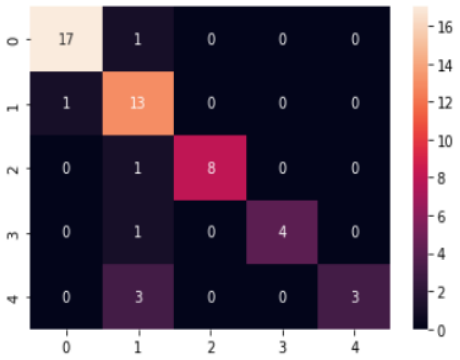

Figure 3: Confusion Matrix for unturned SVC model

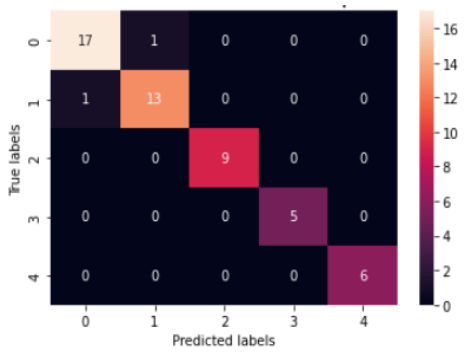

Figure 4: Confusion Matrix for tuned SVC model

The below table illustrates the brain cancer status for four kind of cancer and normal patients are considered in the GE brain cancer database. The confusion matrix value classified with True Positive (TP), True Negative (TN), False Positive (FP) and False Negative (FN).

Brain Cancer Status	Untuned SVC				Tuned SVC		
	TP	TN	FP	FN	TP	TN	FP
Ependymoma	17	33	1	1	17	33	1
Glioblastoma	13	32	6	1	13	37	1
Medulloblastoma	8	43	0	1	9	43	0
Normal	4	47	0	1	5	47	0
Pilocytic astrocytoma	3	46	0	3	6	46	0

Table 1: Relationship status with confusion matrix values

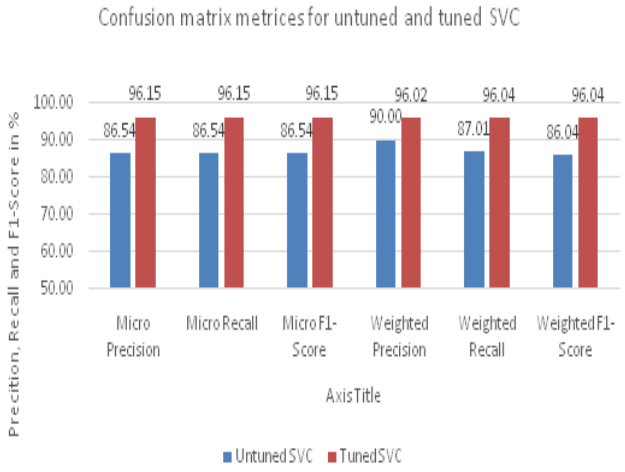

Figure 5: Confusion matrix metrics for unturned and tuned SVC

Figure 5 illustrates the precision, recall and F1-Score for micro as well as weighted and it is used for ordinal type of classification. The severity of brain cancer is determine through number representation and the accuracy of classification is evaluated through Micro precision and micro recall and the average of these two is micro F1-Score. Hence, the value of tuned SVC is trained well through randomized Search CV optimizer whereas the tuned accuracy is micro F1-score as 96.15% is higher than 9.6% of untuned SVC. The information missed has been accumulated through randomized Search CV optimizer as hyperparameter tuning of untuned SVC model. Similarly, the weighted is calculated through support involved in the model which assist in identifying the individual weightage of precision, recall and F1-Score. Moreover, the tuned SVC consumed higher percentage than untuned SVC.

Figure 6 illustrates the true positive rate and false positive rate in which untuned SVC is equal to tuned SVC. The sensitivity of both untuned and tuned SVC is 0.93 and specificity as 0.94. Thus, the optimized played the major role in identifying the weight age and minimize the bias of the SVC model.

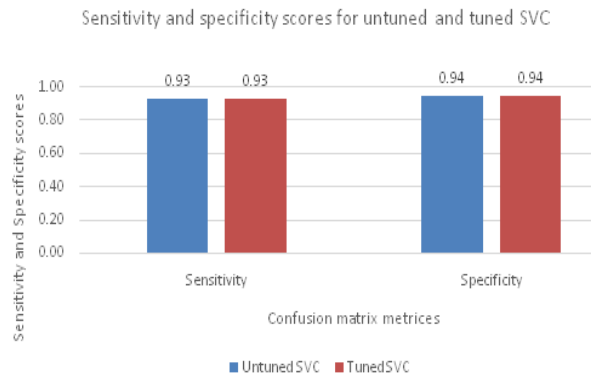

Figure 6: Sensitivity and specificity for untuned and tuned SVC

V. Conclusion

This research has demonstrated to predict brain cancer accurately using GE data with a small number of genes produced through feature extraction by PCA. An effective algorithm has been created for identifying the highly significant genes for brain cancer diagnosis. In improving the untuned balanced accuracy obtained from the lazy predict classifier by hyperparameter tuning algorithm which is explored for top most ML model as SVC. The best model that generates high classification accuracy is SVC has beam improved through optimizer and hyper parameter. The model accuracy is improved trough randomized search CV optimizer with an increase of 9.61% whereas the tuned SVC model accuracy is 96.15% which is more likely to be suitable in predicting the brain cancer classification that assist in earlier detection or severity of brain cancer through GE omnibus.

Reference:

[1.] Andonie, R.; Florea, A.C. Weighted Random Search for CNN Hyperparameter Optimization. Int. J. Comput. Commun. Control 2020, 15.

[2.] Gómez, S., Garrido-Garcia, A., Garcia-Gerique, L., Lemos, I., Suñol, M., de Torres, C., et al. (2018). A Novel Method for Rapid Molecular Subgrouping of Medulloblastoma. Clin. Cancer Res. 24, 1355–1363.

[3.] Liashchynskyi, P.; Liashchynskyi, P. Grid Search, Random Search, Genetic Algorithm: A Big Comparison for NAS. arXiv 2019, arXiv:abs/1912.06059.

[4.] Probst, P.; Boulesteix, A.; Bischl, B. Tunability: Importance of Hyperparameters of Machine Learning Algorithms. J. Mach. Learn. Res. 2019, 20, 53:1–53:32.

[5.] Threshold Methodology to Predict Brain with Gene Expression Pattern by using Machine Learning Algorithm by Y.Tezaaw, Dr.K.Vijaya Lakshmi: International Journal of Novel Research and Development, Volume 8, Issue 4 April 2023 | ISSN: 2456-4184,g136-141.

[6.] Zhang, S., Zeng, T., Hu, B., Zhang, Y. H., Feng, K., Chen, L., et al. (2020). Discriminating Origin Tissues of Tumor Cell Lines by Methylation Signatures and Dys-Methylated Rules. Front. Bioeng. Biotechnol. 8, 507.

[7.] A novel approach to remove pectoral muscle from mediolateral oblique mammograms with hybridization of MSER pectoral and Hough pectoral methods by Ch.Sarada,Dr.KVijayLkashmi,Prof.M.Padma vathamma: Journal of Emerging Technologies and Innovative Research, Volume 9, Issue 8, ISSN:2349-5162

[8.] Pectoral Removal From MLO Mammogram: Ensemble of Adaptive MSER Pectoral and Slope edge detection With Extensive Preprocessing by Ch.Sarada,Dr.KVijay Lkashmi,Prof.M.Padmavathamma: SAMRIDDHI, A journal of physical sciences, Engineering and Technology, Volume 9, Issue 8, August 2022, ISSN-2349-5162

[9.] Metri R, Mohan A, Nsengimana J, et al. Identification of a gene signature for discriminating metastatic from primary melanoma using a molecular interaction network approach. Sci Rep 2017; 7:17314.

[10.] Zhang, S., Zeng, T., Hu, B., Zhang, Y. H., Feng, K., Chen, L., et al. (2020). Discriminating Origin Tissues of Tumor Cell Lines by Methylation Signatures and Dys-Methylated Rules. Front. Bioeng. Biotechnol. 8, 507.

[11.] Gómez, S., Garrido-Garcia, A., Garcia-Gerique, L., Lemos, I., Suñol, M., de Torres, C., et al. (2018). A Novel Method for Rapid Molecular Subgrouping of Medulloblastoma. Clin. Cancer Res. 24, 1355–1363.

[12.] Liashchynskyi, P.; Liashchynskyi, P. Grid Search, Random Search, Genetic Algorithm: A Big Comparison for NAS. arXiv 2019, arXiv:abs/1912.06059.

[13.] Andonie, R.; Florea, A.C. Weighted Random Search for CNN Hyperparameter Optimization. Int. J. Comput. Commun. Control 2020, 15.
[14.] Probst, P.; Boulesteix, A.; Bischl, B. Tunability: Importance of Hyperparameters of Machine Learning Algorithms. J. Mach. Learn. Res. 2019, 20, 53:1–53:32.