ISSN: 2581-3404 (Online) **IF: 5.445 (SJIF)**
International Journal of Innovative Research in Technology and Management, Volume-6, Issue-4, 2022.

A Comparative Study of Machine Learning Algorithms for Predictive Data Analysis

Saumya Tripathi[1], Sweety[2], Rashmi[3], Arpana[4]

Department of CSE (AI & ML) [1]
Department of CSE [2,3,4]
Panipat Institute of Engineering and Technology, Panipat, India [1,2,3,4]

Abstract: *Choosing a suitable learning algorithm is central to predictive data analysis, yet reported rankings often depend on the dataset, the validation protocol and the amount of hyperparameter tuning. This paper presents a reproducible comparison of eight classification algorithms (Logistic Regression, K-Nearest Neighbors, Decision Tree, Naive Bayes, Support Vector Machine, Random Forest, Gradient Boosting and XGBoost) on the Wisconsin Breast Cancer dataset, and nine regression algorithms on the Diabetes progression dataset. All models are evaluated with stratified 10-fold cross-validation and an independent hold-out split, using accuracy, precision, recall, F1-score, ROC-AUC, R^2, RMSE, MAE and training time. On classification, the RBF-kernel SVM and Logistic Regression both reached 97.54% cross-validated accuracy, and grid-search tuning lifted the SVM to 97.72%; a paired test showed no significant difference between the two leaders ($p = 0.209$). Ensemble tree methods did not outperform simpler models, and Decision Tree and Naive Bayes were weakest. On regression, the RBF-kernel SVR achieved the lowest RMSE (54.50) and linear models were close behind, while tree ensembles lagged. The results show that on small, clean tabular data well-regularized simple models are highly competitive.*

Keywords: Machine learning, predictive analytics, classification, regression, cross-validation, comparative study.

Introduction

Predictive data analysis uses historical data to estimate unseen outcomes, and it now underpins decisions in healthcare, finance, security and industry [1]. A large number of supervised learning algorithms are available, ranging from classical linear models and instance-based learners to kernel machines and boosted tree ensembles [12], [13]. Because no single algorithm is best on every problem, practitioners must compare candidates empirically on the data at hand [8].

Published comparisons are difficult to reproduce when they use different validation schemes, report only accuracy, or tune some models more than others [3], [19]. This work therefore fixes one protocol for all models and reports several complementary metrics together with run time. The main contributions are: (i) a unified benchmark of eight classifiers and nine regressors on two standard datasets; (ii) a comparison of default and grid-search-tuned configurations; (iii) a paired statistical test between the top two classifiers; and (iv) an analysis of feature importance and of the accuracy–cost trade-off.

Related Work

Machine learning methods and their applications were surveyed in [1], and the growth of the Python ecosystem used by most practitioners was reviewed in [2]. Kotsiantis [13] compared supervised classification techniques,

while Fernández-Delgado et al. [8] evaluated 179 classifiers on 121 datasets and found that Random Forest and SVM variants performed best on average. Gradient boosting [17] has been extended into efficient systems such as XGBoost [6], LightGBM [5] and CatBoost [4], which are frequently dominant on large tabular data. Deep learning [7] excels on images, speech and text but is less often advantageous on small tabular datasets. The influence of hyperparameters has also been studied: Bergstra and Bengio [10] showed that random search is an efficient alternative to grid search, and Probst et al. [3] quantified how tunable different algorithms are. Demšar [14] recommended statistical tests for classifier comparison. The present study follows these recommendations with respect to resampling, tuning and significance testing.

Methodology



Fig. 1. Workflow of the proposed comparative study.

Datasets

Two public datasets bundled with scikit-learn [11] were used (Table I). The Wisconsin Diagnostic Breast Cancer dataset contains 30 features computed from digitized images of fine-needle aspirates, with the task of separating malignant from benign tumors [20]. The Diabetes dataset has ten baseline physiological variables and a quantitative measure of disease progression one year later [15].

TABLE I: DATASET DESCRIPTION

Property	**Breast Cancer**	**Diabetes**
Task	Classification	Regression
Instances	569	442
Features	30 (numeric)	10 (numeric)
Target	Malignant / Benign	Disease progression
Class balance	212 / 357	Continuous
Source	UCI / scikit-learn	Efron et al. [15]

B. Algorithms

The classifiers were Logistic Regression (LR) [9], K-Nearest Neighbors (KNN, k = 7) [22], Decision Tree (DT) [21], Gaussian Naive Bayes (NB), SVM with an RBF kernel [18], Random Forest (RF, 200 trees) [16], Gradient

Boosting (GB) [17] and XGBoost (XGB) [6]. The regressors were Linear, Ridge and Lasso regression, KNN, DT, RBF-kernel SVR, RF, GB and XGB [12].

C. Preprocessing and Evaluation Protocol

All features were standardized to zero mean and unit variance inside a pipeline, so the scaler was fitted only on training folds and no information leaked from validation folds [19]. Classification used stratified 10-fold cross-validation with a fixed random seed, reporting accuracy, precision, recall, F1-score, ROC-AUC and mean fit time. Regression used 10-fold cross-validation with R^2, RMSE and MAE. A separate stratified 75/25 hold-out split (426 training, 143 test samples) was used for ROC curves and confusion matrices.

D. Hyperparameter Tuning and Significance Testing

Grid search with 5-fold inner cross-validation was applied to SVM, RF, XGB and KNN [3], [10], and the chosen configuration was then re-evaluated with the outer 10-fold protocol. The two best classifiers were compared using 5×10 repeated stratified cross-validation (50 paired scores) with a paired t-test and a Wilcoxon signed-rank test [14]. Experiments were implemented in Python using scikit-learn [11] and XGBoost [6].

Result and Discussion

A. Classification Performance

Table II lists the cross-validated metrics. SVM and LR were the top models, each with 97.54% accuracy; SVM had the highest ROC-AUC (99.56%) and LR the highest hold-out accuracy (98.60%). XGB (96.67%) and KNN (96.49%) followed, then GB (95.96%) and RF (95.44%). NB (93.15%) and DT (92.62%) performed worst, DT also having the lowest AUC (92.00%) because a single tree produces coarse probability estimates. Fig. 2 compares accuracies with standard deviation error bars.

TABLE II: CROSS-VALIDATED CLASSIFICATION RESULTS (%) — BREAST CANCER

Model	Acc.	Prec.	Rec.	F1	AUC
LR	97.54	97.36	98.88	98.08	99.47
KNN	96.49	95.55	99.16	97.28	98.61
DT	92.62	94.07	94.39	94.10	92.00
NB	93.15	94.20	95.26	94.58	98.67
SVM	97.54	97.84	98.33	98.05	99.56
RF	95.44	96.05	96.92	96.41	99.07
GB	95.96	96.00	97.75	96.83	99.18
XGB	96.67	96.81	98.05	97.38	99.38

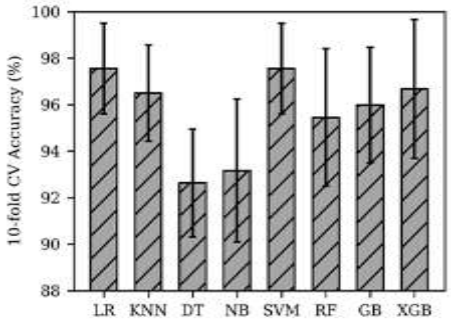


Fig. 2. Ten-fold cross-validated accuracy of the classifiers (error bars: ±1 SD).

Table III reports stability and cost. The accuracy standard deviation ranged from 1.95 to 3.08 points across models, so differences of a few tenths of a percent between the leaders are within normal fold-to-fold variation. LR, KNN and NB trained in under 10 ms, SVM in about 24 ms, while RF, GB and XGB needed 168–587 ms, roughly 7 to 25 times the SVM cost without any accuracy benefit on this dataset.

TABLE III: STABILITY, HOLD-OUT ACCURACY AND TRAINING COST

Model	**Acc. SD (%)**	**Hold-out Acc. (%)**	**Train time (ms)**
LR	1.95	98.60	8.0
KNN	2.07	97.90	3.1
DT	2.33	92.31	12.0
NB	3.08	93.71	3.8
SVM	1.95	97.90	23.7
RF	2.96	95.80	447.9
GB	2.49	95.80	586.5
XGB	2.98	96.50	168.5

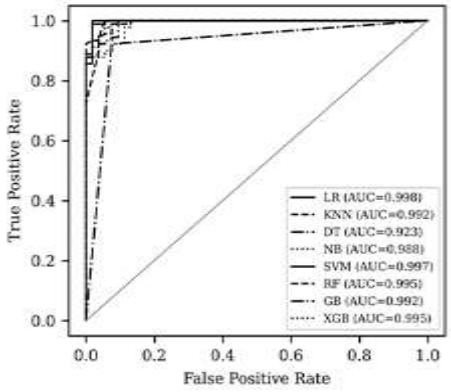


Fig. 3. ROC curves of all classifiers on the 25% hold-out set.

Fig. 3 shows the ROC curves on the hold-out set; all models other than DT are close to the upper-left corner, which indicates strong class separability in this dataset. Fig. 4 gives the SVM confusion matrix on the hold-out set: 52 malignant and 88 benign cases were classified correctly, with one malignant tumor predicted benign (false negative) and two benign tumors predicted malignant (false positive), a hold-out accuracy of 97.90%. In screening, false negatives are the costlier error, so recall of the malignant class should be monitored in addition to accuracy.

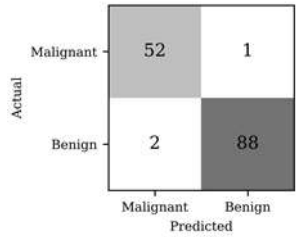


Fig. 4. Confusion matrix of the SVM on the hold-out set.

B. Effect of Hyperparameter Tuning

Tuning (Table IV) gave small and uneven changes. SVM improved from 97.54% to 97.72% with C = 10 and γ = 0.01, whereas RF and XGB changed by only 0.17 and −0.71 points, and KNN with k = 9 was unchanged. The slight drop for XGB shows that a configuration selected by inner cross-validation can be marginally worse on the outer folds, a reminder that tuning gains on small datasets are close to noise [3].

TABLE IV: ACCURACY (%) BEFORE AND AFTER GRID SEARCH

Model	**Best parameters**	**Acc. before**	**Acc. after**
SVM	C=10, gamma=0.01	97.54	97.72
RF	max_depth=None, n_estimators=100	95.44	95.61
XGB	learning_rate=0.1, max_depth=3, n_estimators=100	96.67	95.96
KNN	n_neighbors=9	96.49	96.49

C. Statistical Significance

Over 50 paired cross-validation scores, SVM scored 97.43% and LR 97.78%. The difference was not significant under the paired t-test ($p = 0.209$) or the Wilcoxon test ($p = 0.303$). Hence the two models cannot be ranked reliably, and the simpler and faster LR is a sound choice here [14].

D. Feature Importance

Random Forest importances (Fig. 5, Table V) show that the “worst” (largest) values of perimeter, area and concave points, together with mean concave points, carry most of the predictive signal, about 50% of total importance in the top four features. This agrees with the clinical expectation that large, irregular nuclei indicate malignancy and makes the model outputs easier to explain.

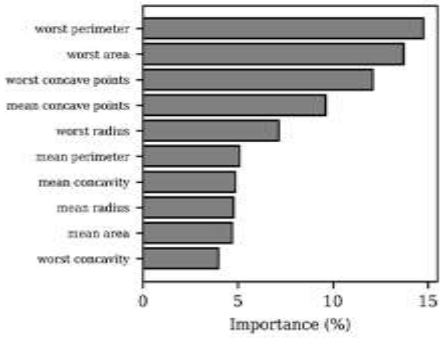


Fig. 5. Top ten features by Random Forest importance.

TABLE V: MOST IMPORTANT FEATURES

Rank	**Feature (Random Forest importance, %)**
1	worst perimeter (14.76)
2	worst area (13.73)
3	worst concave points (12.09)
4	mean concave points (9.62)
5	worst radius (7.16)
6	mean perimeter (5.08)

E. Regression Performance

Table VI summarizes the Diabetes experiments. RBF-kernel SVR gave the best result (R^2 = 0.474, RMSE = 54.50, MAE = 43.95), closely followed by Lasso, Ridge and Linear regression ($R^2 \approx 0.465$, RMSE ≈ 54.8). KNN, RF, GB and XGB had RMSE between 56.4 and 58.0, and DT was worst (RMSE 62.59, R^2 = 0.295). Fig. 6 compares the RMSE values, and Fig. 7 plots the SVR predictions against actual values, showing that the model captures the general trend but underestimates extreme progression scores.

TABLE VI: TEN-FOLD CROSS-VALIDATED REGRESSION RESULTS — DIABETES

Model	**R^2**	**RMSE**	**MAE**
LinReg	0.465	54.86	44.48
Ridge	0.465	54.85	44.47
Lasso	0.466	54.83	44.44
KNN	0.404	57.98	46.51
DT	0.295	62.59	50.55
SVR	0.474	54.50	43.95
RF	0.413	57.54	46.86
GB	0.432	56.45	46.34
XGB	0.425	56.78	46.25

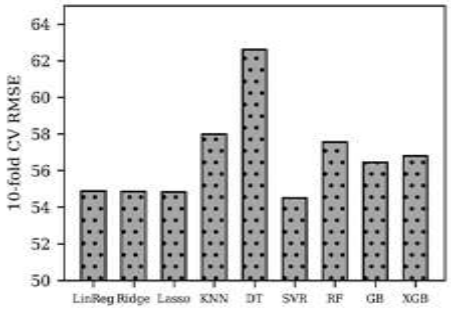


Fig. 6. Cross-validated RMSE of the regression models (lower is better).

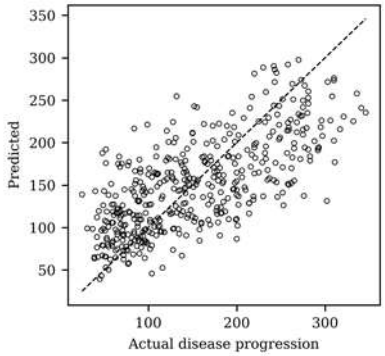


Fig. 7. Predicted versus actual values for the best regressor (SVR), from cross-validation.

F. Discussion and Limitations

Three observations follow. First, ensemble tree methods, often dominant on large datasets [5], [6], [8], were not superior on these small, clean, mostly linearly separable datasets; regularized linear and kernel models matched or beat them at a fraction of the cost. Second, the top classifiers differ by less than the fold-to-fold variation, so rankings drawn from a single run should be treated with care [14]. Third, the regression problem is inherently noisy, and even the best model explains under half of the variance.

Limitations are that only two small datasets were used, the grid searches were modest, and class imbalance, missing values and categorical features were not studied. Conclusions should therefore not be generalized to large or heterogeneous data without further testing.

Conclusion

This paper compared eight classifiers and nine regressors under one validation protocol. SVM and Logistic Regression led on breast cancer classification (97.54% accuracy; 97.72% for the tuned SVM) with no significant difference between them, and SVR led on diabetes progression with an RMSE of 54.50. Tree ensembles brought no gain and higher computational cost. The findings support the practice of starting with strong simple baselines and justifying complex models only when they show a significant improvement. Future work will extend the benchmark to larger and imbalanced datasets, add LightGBM, CatBoost and neural networks [4], [5], [7], use Bayesian or random search for tuning [10], and include model explanation methods.



References

[1] I. H. Sarker, "Machine learning: Algorithms, real-world applications and research directions," SN Comput. Sci., vol. 2, no. 3, p. 160, 2021.
[2] S. Raschka, J. Patterson, and C. Nolet, "Machine learning in Python: Main developments and technology trends in data science, machine learning, and artificial intelligence," Information, vol. 11, no. 4, p. 193, 2020.
[3] P. Probst, A.-L. Boulesteix, and B. Bischl, "Tunability: Importance of hyperparameters of machine learning algorithms," J. Mach. Learn. Res., vol. 20, no. 53, pp. 1–32, 2019.
[4] L. Prokhorenkova, G. Gusev, A. Vorobev, A. V. Dorogush, and A. Gulin, "CatBoost: Unbiased boosting with categorical features," in Proc. Adv. Neural Inf. Process. Syst. (NeurIPS), vol. 31, 2018, pp. 6638–6648.
[5] G. Ke et al., "LightGBM: A highly efficient gradient boosting decision tree," in Proc. Adv. Neural Inf. Process. Syst. (NIPS), vol. 30, 2017, pp. 3146–3154.
[6] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining, 2016, pp. 785–794.
[7] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," Nature, vol. 521, no. 7553, pp. 436–444, 2015.
[8] M. Fernández-Delgado, E. Cernadas, S. Barro, and D. Amorim, "Do we need hundreds of classifiers to solve real world classification problems?," J. Mach. Learn. Res., vol. 15, pp. 3133–3181, 2014.
[9] D. W. Hosmer, S. Lemeshow, and R. X. Sturdivant, Applied Logistic Regression, 3rd ed. Hoboken, NJ, USA: Wiley, 2013.
[10] J. Bergstra and Y. Bengio, "Random search for hyper-parameter optimization," J. Mach. Learn. Res., vol. 13, pp. 281–305, 2012.
[11] F. Pedregosa et al., "Scikit-learn: Machine learning in Python," J. Mach. Learn. Res., vol. 12, pp. 2825–2830, 2011.
[12] T. Hastie, R. Tibshirani, and J. Friedman, The Elements of Statistical Learning, 2nd ed. New York, NY, USA: Springer, 2009.
[13] S. B. Kotsiantis, "Supervised machine learning: A review of classification techniques," Informatica, vol. 31, no. 3, pp. 249–268, 2007.
[14] J. Demšar, "Statistical comparisons of classifiers over multiple data sets," J. Mach. Learn. Res., vol. 7, pp. 1–30, 2006.
[15] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani, "Least angle regression," Ann. Statist., vol. 32, no. 2, pp. 407–499, 2004.
[16] L. Breiman, "Random forests," Mach. Learn., vol. 45, no. 1, pp. 5–32, 2001.
[17] J. H. Friedman, "Greedy function approximation: A gradient boosting machine," Ann. Statist., vol. 29, no. 5, pp. 1189–1232, 2001.
[18] C. Cortes and V. Vapnik, "Support-vector networks," Mach. Learn., vol. 20, no. 3, pp. 273–297, 1995.
[19] R. Kohavi, "A study of cross-validation and bootstrap for accuracy estimation and model selection," in Proc. 14th Int. Joint Conf. Artif. Intell. (IJCAI), 1995, pp. 1137–1143.
[20] W. N. Street, W. H. Wolberg, and O. L. Mangasarian, "Nuclear feature extraction for breast tumor diagnosis," in Proc. IS&T/SPIE Int. Symp. Electron. Imaging, vol. 1905, 1993, pp. 861–870.
[21] J. R. Quinlan, "Induction of decision trees," Mach. Learn., vol. 1, no. 1, pp. 81–106, 1986.
[22] T. Cover and P. Hart, "Nearest neighbor pattern classification," IEEE Trans. Inf. Theory, vol. 13, no. 1, pp. 21–27, 1967.