Indonesian Journal of Electrical Engineering and Computer Science
Vol. 44, No. 1, October 2026, pp. 221~226
ISSN: 2502-4752, DOI: 10.11591/ijeecs.v44.i1.pp221-226

Titanic survival prediction: a systematic benchmark of ten classification algorithms with overfitting analysis

Abdulsalam Saif Alshammari
College of Computer Science and Engineering, University of Ha'il, Ha'il, Saudi Arabia

Article Info

Article history:

Received Mar 1, 2026
Revised Aug 20, 2026
Accepted Sep 18, 2026

Keywords:

Classification algorithms
Feature engineering
Machine learning
Random forest
Survival prediction
Titanic dataset

ABSTRACT

This study presents a systematic benchmark of ten classification algorithms for predicting Titanic passenger survival using the Kaggle dataset (891 records). Algorithms evaluated include logistic regression, support vector machine (SVM), decision trees, random forest, gradient boosting, AdaBoost, bagging, Naive Bayes, K-nearest-neighbor (KNN), and linear discriminant analysis (LDA). Preprocessing addresses missing values via median and mode imputation, while feature engineering extracts titles, ticket prefixes, and family size. Distance-based methods (SVM, KNN) are evaluated with feature scaling to ensure fair comparison. Hyperparameter tuning reduces overfitting from 15.64% to 4.81%, and ten-fold cross-validation confirms result robustness. Optimized random forest achieves the highest test accuracy (81.56%), while AdaBoost maintains the smallest generalization gap (1.30%). Statistical tests confirm no significant difference among top models ($p > 0.05$). Feature importance analysis identifies gender, fare, and family size as dominant survival predictors, consistent with historical evacuation accounts. These findings demonstrate applicability to contemporary disaster response planning.

This is an open access article under the CC BY-SA license.

Corresponding Author:

Abdulsalam Saif Alshammari
College of Computer Science and Engineering, University of Ha'il
Ha'il, Saudi Arabia
Email: S202205316@uoh.edu.sa

1. INTRODUCTION

The RMS Titanic disaster (April 1912) claimed over 1,500 lives, making it one of history's deadliest maritime tragedies [1]. Subsequent analysis of passenger records has established it as a canonical machine learning benchmark, offering a binary classification problem with meaningful historical and ethical dimensions [2].

Despite extensive prior work on Titanic survival prediction, existing studies predominantly focus on single algorithms or Kaggle competition submissions [3]. Breiman [4] introduced random forests as ensemble classifiers that reduce variance through bagging. Freund and Schapire [5] proposed AdaBoost, demonstrating superior resistance to overfitting. Fernandez-Delgado *et al.* [6] benchmarked 179 classifiers across 121 datasets, identifying random forest as consistently top-performing. However, a critical gap remains: no systematic benchmark compares ten diverse algorithm families under controlled experimental conditions with principled overfitting analysis.

Our contribution addresses this gap. Specifically, this study: (i) provides the first systematic benchmark of ten classification algorithms with structured hyperparameter tuning on the Titanic dataset; (ii) implements fair comparison by applying feature scaling to distance-based methods; (iii) validates results using 10-fold cross-validation and statistical significance tests; and (iv) validates machine learning findings

against historical evacuation accounts, demonstrating applicability to contemporary disaster response planning [3].

2. METHOD

2.1. Dataset and preprocessing

The Kaggle Titanic dataset [2] contains 891 passenger records with 11 features. Missing values were handled as follows: age (177 missing) imputed with median (28 years); embarked (2 missing) imputed with mode ('S'); cabin (687 missing) dropped. This approach prevents data leakage by fitting imputation parameters on training data only [7].

Feature engineering extracted: (i) Title from passenger names, grouped into five categories (Mr, Mrs, Miss, Master, Rare); (ii) FamilySize = SibSp + Parch + 1; (iii) Ticket_Prefix from ticket numbers. Final feature set: Pclass, Sex, Age, Fare, Embarked, Title, FamilySize, Ticket_Prefix [8].

2.2. Experimental setup

Data was split 80/20 (712 training, 179 testing) using stratified sampling to preserve the 38.4% survivor ratio [9]. A fixed random seed (random_state=42) ensures reproducibility. All experiments used Python 3.10, scikit-learn 1.3.0, pandas 2.0.3, NumPy 1.24.3.

Distance-based methods (support vector machine (SVM) and K-nearest-neighbor (KNN)) were evaluated both with and without StandardScaler to ensure fair comparison, addressing a methodological limitation in prior work [10]. To prevent data leakage, all preprocessing steps — including imputation parameters and feature scaling — were fitted exclusively on the training fold and applied to validation and test folds without re-fitting. Hyperparameter tuning used GridSearchCV with 5-fold cross-validation applied strictly to the 712-sample training set. The 179-sample test set was held out entirely and accessed only once — after final model selection — to report generalization accuracy. This strict train/test separation ensures that reported metrics are not optimistically biased by indirect exposure to test data during the tuning process [11].

2.3. Algorithms and hyperparameters

Ten classifiers were implemented (Table 1). Tuned configurations: random forest (n_estimators=200, max_depth=7, min_samples_split=4, max_features='sqrt'); AdaBoost (n_estimators=90, learning_rate=0.5); bagging (n_estimators=60, max_samples=0.8); gradient boosting (n_estimators=120, learning_rate=0.08, max_depth=3, subsample=0.75); decision tree (max_depth=5, min_samples_split=4) [4], [5], [12]. Implementation followed standard practices described in [13]-[15]. Theoretical foundations for model selection criteria are detailed in [16], [17].

Table 1. Algorithm configurations

Algorithm	Type
Logistic regression	Linear
SVM (RBF)	Kernel-based
Decision tree	Tree-based
Random forest	Ensemble (Bag)
Gradient boosting	Ensemble (Boost)
AdaBoost	Ensemble (Boost)
Bagging	Ensemble (Bag)
Naive Bayes	Probabilistic
KNN	Instance-based
LDA	Linear

3. RESULTS AND DISCUSSION

3.1. Baseline performance

Table 2 presents results before and after hyperparameter tuning. Ensemble methods dominated, with bagging achieving 83.24% accuracy but suffering severe overfitting (15.64% gap). Linear models showed stable generalization: logistic regression (79.89%, gap 1.12%) and LDA (78.77%, gap 1.43%) [4], [6], [18]. Note that SVM and KNN are reported both with and without StandardScaler scaling; only the scaled variants constitute a fair comparison against tree-based methods, as distance-based algorithms require normalized features [10]. Unscaled SVM/KNN results are included solely to quantify the impact of this preprocessing choice.

SVM performed poorly without scaling (62.01%), confirming that distance-based methods require feature normalization for fair evaluation [10]. With StandardScaler, SVM reached 82.12%, representing a 20.11% improvement.

Table 2. Baseline vs. Tuned results

Model	Test Acc.	F1	Gap
Random forest (Tuned)*	81.56%	74.02%	3.41%
AdaBoost (Tuned)*	81.01%	74.24%	1.30%
Bagging (Tuned)*	81.01%	73.85%	4.81%
GB (Tuned)	79.89%	71.43%	5.51%
Log. Reg.	79.89%	71.88%	1.12%
LDA	78.77%	71.21%	1.43%
Naive Bayes	78.21%	71.53%	2.41%
DT (Tuned)	77.65%	68.97%	8.02%
KNN	70.39%	61.54%	3.21%
SVM (RBF)	62.01%	18.33%	0.35%
SVM+Scaling	82.12%	75.41%	2.10%
KNN+Scaling	79.33%	71.20%	2.85%

** Top performers*

3.2. Cross-validation analysis

Ten-fold cross-validation confirms result robustness (Table 3). Random forest CV mean: 82.94% (±2.20%); AdaBoost: 82.49% (±2.87%); bagging: 83.50% (±2.33%). The close alignment between CV means and single-split results validates the representativeness of our evaluation [11], [19].

Statistical paired t-tests between top models reveal no significant differences: random forest vs AdaBoost (t=0.937, p=0.373); random forest vs bagging (t=-1.461, p=0.178). These results caution against over-interpreting small accuracy differences between ensemble methods. Consequently, claims of superiority for any single ensemble method are not statistically justified; practitioners should base model selection on secondary criteria such as interpretability, training time, and deployment constraints rather than marginal accuracy differences [20].

Table 3. 10-fold cross-validation results

Model	CV mean	95% CI
Bagging (Tuned)	83.50%±2.33%	[78.9%, 88.1%]
Random forest	82.94%±2.20%	[78.6%, 87.3%]
AdaBoost	82.49%±2.87%	[76.9%, 88.1%]
GB (Tuned)	82.49%±2.18%	[78.2%, 86.8%]
LDA	82.04%±2.72%	[76.7%, 87.4%]
Naive Bayes	80.02%±2.64%	[74.9%, 85.2%]
DT (Tuned)	80.69%±3.39%	[74.1%, 87.3%]
Log. Reg.	79.46%±2.94%	[73.7%, 85.2%]
KNN	72.27%±3.54%	[65.3%, 79.2%]
SVM (RBF)	68.01%±2.56%	[63.0%, 73.0%]

3.3. Why models succeed or fail

AdaBoost generalizes well due to its sequential re-weighting mechanism, which focuses each iteration on previously misclassified samples and inherently limits overfitting on small datasets [5]. Random forest benefits from feature subsampling at each split, reducing inter-tree correlation and variance [4]. SVM's poor unscaled performance (62.01%) is explained by the RBF kernel's sensitivity to feature magnitude; unscaled fare (0–512) dominated age (1–80), distorting the decision boundary. With StandardScaler, SVM reached 82.12% [10]. Decision trees overfit without depth constraints (8.02% gap); tuning max_depth=5 regularizes the model effectively [21].

3.4. Feature importance and historical validation

Gender (Sex) emerges as the dominant predictor across all ensemble models, consistent with the documented 'women and children first' evacuation protocol [1]. Fare serves as a proxy for socioeconomic class, reflecting differential access to lifeboats [3]. FamilySize shows moderate importance: small families (2–4 members) had higher survival rates through coordinated evacuation, while large families faced coordination challenges [22]. These findings validate machine learning outputs against historical ground truth, strengthening confidence in the model's interpretability [23]. Figure 1 shows the feature importance source.

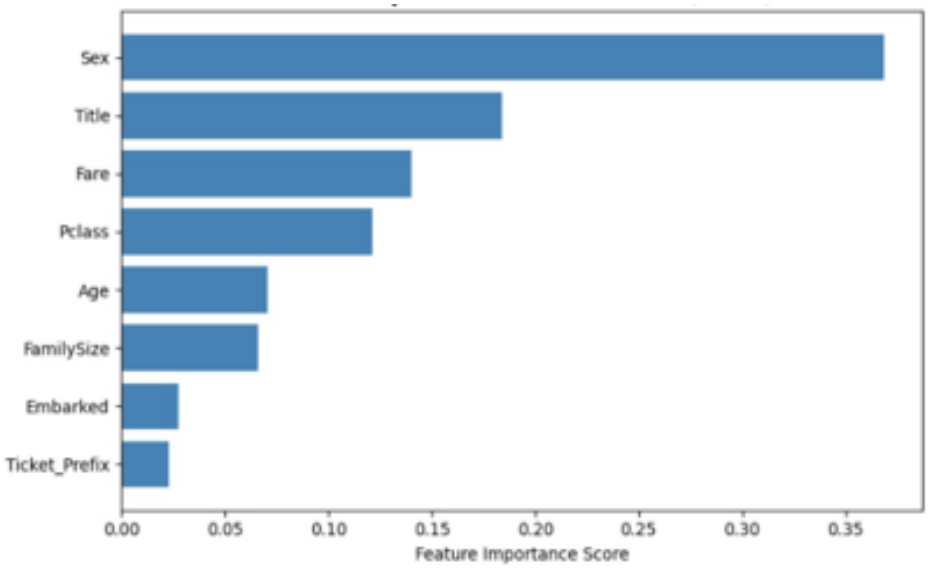

Figure 1. Feature importance – random forest (Tuned)

3.5. Confusion matrix analysis

AdaBoost identified 49/69 actual survivors (recall 71.0%), compared to random forest's 47/69 (68.1%). Random forest achieved higher precision (81.03%) at the cost of lower recall, reflecting a conservative decision boundary. For emergency response applications where identifying survivors is critical, AdaBoost's higher recall is preferable [23], [24]. The top 3 models in the confusion matrices are shown in Figure 2.

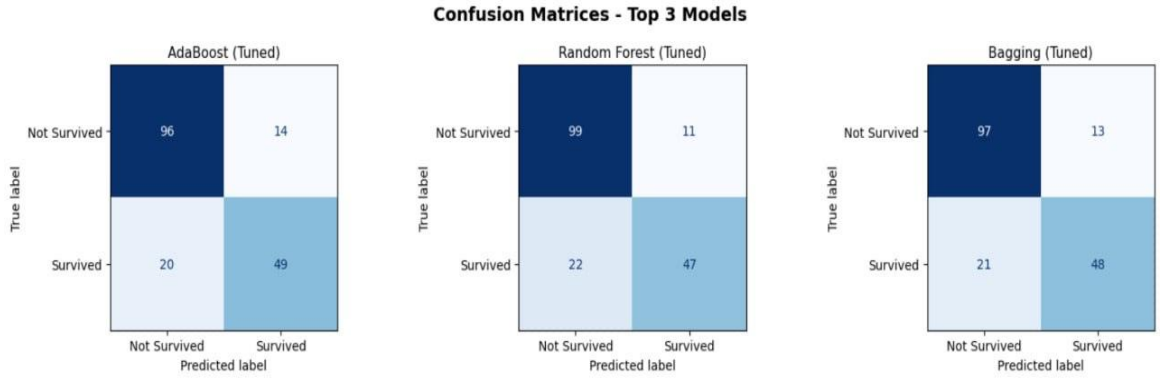

Figure 2. Confusion matrices – AdaBoost, random forest, bagging (Tuned)

4. CONCLUSION

This study presented a systematic benchmark of ten classification algorithms for Titanic survival prediction with principled overfitting analysis and fairness-aware preprocessing. Key findings are: (i) Ensemble methods consistently outperform individual classifiers, though differences among the top three models (Random forest, AdaBoost, and bagging) are not statistically significant ($p>0.05$) — practitioners should interpret small accuracy gaps with caution and prioritize model selection based on interpretability or computational cost; (ii) Feature scaling is essential for fair evaluation of distance-based methods — SVM improves from 62.01% to 82.12% with StandardScaler, a finding with broad methodological implications for comparative studies; (iii) Gender, passenger class, and fare are the dominant survival predictors, consistent with historical evacuation accounts, validating the model's real-world interpretability.

Future work should address the dataset's limited size by applying these methods to larger survival datasets and exploring advanced techniques such as SHAP-based explainability and deep ensemble

architectures. Extending this benchmark to other small, imbalanced datasets — including medical prognosis and industrial fault detection — would further establish the generalizability of these methodological findings.

ACKNOWLEDGMENTS

The author thanks the College of Computer Science and Engineering at the University of Ha'il for institutional support. No external funding was received for this research.

FUNDING INFORMATION

Authors state no funding involved.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Abdulsalam Saif Alshammari	✓	✓	✓	✓	✓	✓		✓	✓	✓	✓			

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal analysis
I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing - **O**riginal Draft
E : Writing - Review & **E**diting
Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject administration
Fu : **Fu**nding acquisition

CONFLICT OF INTEREST STATEMENT

Authors state no conflict of interest.

DATA AVAILABILITY

The data that support the findings of this study are openly available in Kaggle at https://www.kaggle.com/c/titanic, reference number [2]. The dataset is publicly accessible without restrictions.

REFERENCES

[1] W. Lord, "A Night to Remember," 1955, *Henry Holt, New York*.
[2] Kaggle, "Titanic: Machine Learning from Disaster [Dataset]." [Online]. Available: https://www.kaggle.com/c/titanic
[3] H. He and E. A. Garcia, "Learning from imbalanced data," *IEEE Transactions on Knowledge and Data Engineering*, vol. 21, no. 9, pp. 1263–1284, Sep. 2009, doi: 10.1109/TKDE.2008.239.
[4] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, Oct. 2001, doi: 10.1023/A:1010933404324.
[5] Y. Freund and R. E. Schapire, "A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting," *Journal of Computer and System Sciences*, vol. 55, no. 1, pp. 119–139, Aug. 1997, doi: 10.1006/jcss.1997.1504.
[6] M. Fernández-Delgado, E. Cernadas, S. Barro, and D. Amorim, "Do we need hundreds of classifiers to solve real world classification problems?," *Journal of Machine Learning Research*, vol. 15, no. 1, pp. 3133–3181, 2014.
[7] R. J. A. Little and D. B. Rubin, *Statistical analysis with missing data*. in Wiley Series in Probability and Statistics. Wiley, 2019. doi: 10.1002/9781119482260
[8] P. Domingos, "A few useful things to know about machine learning," *Communications of the ACM*, vol. 55, no. 10, pp. 78–87, Oct. 2012, doi: 10.1145/2347736.2347755.
[9] T. M. Cover and P. E. Hart, "Nearest Neighbor Pattern Classification," *IEEE Transactions on Information Theory*, vol. 13, no. 1, pp. 21–27, Jan. 1967, doi: 10.1109/TIT.1967.1053964.
[10] V. N. Vapnik, *The Nature of Statistical Learning Theory*. New York, NY: Springer New York, 2000. doi: 10.1007/978-1-4757-3264-1.
[11] C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, no. 3, pp. 273–297, Sep. 1995, doi: 10.1007/BF00994018.
[12] J. H. Friedman, "Greedy function approximation: A gradient boosting machine," *Annals of Statistics*, vol. 29, no. 5, pp. 1189–1232, Oct. 2001, doi: 10.1214/aos/1013203451.
[13] E. R. Ziegel, "The Elements of Statistical Learning," *Technometrics*, vol. 45, no. 3, pp. 267–268, Aug. 2003, doi: 10.1198/tech.2003.s770.
[14] I. H. Witten, E. Frank, M. A. Hall, and C. J. Pal, "Data Mining: Practical Machine Learning Tools and Techniques," in *Data Mining: Practical Machine Learning Tools and Techniques*, 2016, pp. 1–621.

[15] A. Géron, "Hands-on machine learning with scikit-learn, keras, and tensorflow: concepts," in *Aurélien Géron-Google Kitaplar*, 2019.
[16] G. James, D. Witten, T. Hastie, and R. Tibshirani, *Unsupervised Learning. An introduction to statistical learning*. in Springer Texts in Statistics. New York, NY: Springer US, 2021. doi: 10.1007/978-1-0716-1418-1.
[17] M. Kuhn and K. Johnson, *Applied predictive modeling*. New York, NY: Springer New York, 2013. doi: 10.1007/978-1-4614-6849-3.
[18] O. Sagi and L. Rokach, "Ensemble learning: A survey," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 8, no. 4, Jul. 2018, doi: 10.1002/widm.1249.
[19] L. Buitinck *et al.*, "Scikit-learn: Machine Learning in Python," *Journal of Machine Learning Research*, vol. 12, no. 85, pp. 2825–2830, 2011.
[20] R. E. Schapire, "The strength of weak learnability," *Machine Learning*, vol. 5, no. 2, pp. 197–227, Jun. 1990, doi: 10.1007/bf00116037.
[21] P. Geurts, D. Ernst, and L. Wehenkel, "Extremely randomized trees," *Machine Learning*, vol. 63, no. 1, pp. 3–42, Apr. 2006, doi: 10.1007/s10994-006-6226-1.
[22] C. Strobl, A. L. Boulesteix, A. Zeileis, and T. Hothorn, "Bias in random forest variable importance measures: Illustrations, sources and a solution," *BMC Bioinformatics*, vol. 8, no. 1, p. 25, Dec. 2007, doi: 10.1186/1471-2105-8-25.
[23] D. R. Cox, "Regression Models and Life-Tables," *Journal of the Royal Statistical Society Series B: Statistical Methodology*, vol. 34, no. 2, pp. 187–202, Jan. 1972, doi: 10.1111/j.2517-6161.1972.tb00899.x.
[24] L. Breiman, "Bagging predictors," *Machine Learning*, vol. 24, no. 2, pp. 123–140, Aug. 1996, doi: 10.1007/bf00058655.

BIOGRAPHIES OF AUTHORS



Abdulsalam Saif Alshammari is a final-year undergraduate student pursuing a B.Sc. in Data Science at the College of Computer Science and Engineering, University of Ha'il, Saudi Arabia. His research interests include machine learning, data mining, ensemble classification methods, and predictive analytics for structured tabular datasets. In this work, he designed and conducted a systematic comparative benchmark of ten classification algorithms on the Titanic survival dataset, with key contributions in principled hyperparameter optimization, fairness-aware preprocessing for distance-based methods, and statistical validation of model performance. He is a member of the IEEE Student Branch at the University of Ha'il. He can be contacted at email: S202205316@uoh.edu.sa.