WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal Homepage: www.ijrpr.com ISSN 2582-7421

Evaluating Call Duration Availability in Bank Marketing Prediction

Arjun Ananthakrishnan, S Sreya and Dr. G. Paavai Anand, *Dr. G. Paavai Anand

Undergraduate Student, *Faculty Guide

B.Tech in Computer Science, *Department of Computer Science and Engineering, SRM Institute of Science and Technology, Vadapalani Campus, India*
*Faculty Guide, *Department of Computer Science and Engineering, SRM Institute of Science and Technology, Vadapalani Campus, India*
Emails: aa0509@srmist.edu.in, ss7682@srmist.edu.in

ABSTRACT

Call duration can improve retrospective bank marketing prediction while being unavailable when a customer is selected for a call. This study quantifies that distinction through a matched comparison of four classification pipelines on the original Bank Marketing dataset. Logistic Regression, Decision Tree, Random Forest and K Nearest Neighbors were trained with all 16 predictors and with duration removed. The 45,211 real records were divided into 28,934 fitting, 7,234 validation and 9,043 test records using shared stratified indices. Preprocessing was fitted within each training pipeline, and the without-duration candidate was selected by validation F1. Random Forest achieved the highest candidate F1 in both conditions, decreasing from 0.5862 with duration to 0.4569 without it; ROC AUC decreased from 0.9240 to 0.7971. After refitting the selected without-duration forest on 36,168 development records, test F1 was 0.4549, ROC AUC was 0.8001 and average precision was 0.4470, with precision 0.3725 and recall 0.5841. The always-no baseline achieved higher accuracy, 0.8830, but zero subscriber recall. The results show how feature availability and evaluation criteria alter the interpretation of an otherwise conventional benchmark. They support a reproducible applied baseline rather than a new algorithm or a fully validated pre-call system. Remaining last-contact fields, one random split and the absence of external validation constrain operational conclusions.

Keywords: bank marketing; feature availability; call duration; classification; class imbalance; average precision; random forest; reproducibility

1. Introduction

A predictive model for bank telemarketing is useful only in relation to a specified decision. Selecting customers before contact, estimating the response of a completed call and describing historical campaign outcomes are different tasks, even when they share a binary subscription label. A feature recorded after a decision may be informative in a historical dataset without being admissible for the intended prediction. The Bank Marketing application studied by Moro, Cortez and Rita provides a relevant setting for examining that distinction (Moro, Cortez, & Rita, 2014).

The original bank-full.csv data contain customer attributes, campaign history and fields associated with the most recent contact (Moro, Rita, & Cortez, 2014). Duration is particularly important for task definition because the length of a call is known only after that call has occurred. Including it in a retrospective comparison is legitimate if the question is explicitly retrospective. Interpreting the resulting score as evidence of a model that can select customers before the same call, however, changes the prediction task without changing the evaluation. The issue is information availability, not necessarily a literal copy of the target label.

A second problem is metric interpretation. Subscribers form only 11.70% of the records in the executed dataset. A model can therefore achieve high accuracy while detecting few, or none, of the positive outcomes. Precision-recall analysis is useful in such settings because it makes the correctness of positive predictions and the coverage of actual positives explicit (Saito & Rehmsmeier, 2015). An evaluation centered on overall accuracy may select a different model from one centered on F1 or ranking quality.

The present study addresses three questions. First, how do the implemented classifiers change when duration is removed while the training and evaluation records are held fixed? Second, does the identity of the apparent best model depend on whether accuracy, F1 or ranking metrics are considered? Third, what performance does the validation-selected without-duration pipeline achieve after refitting on all available development records? These questions are answered using the completed notebook and exported artifacts, rather than synthetic smoke-test outcomes.

The contribution is a transparent empirical comparison with an explicit separation between candidate evaluation, validation-based selection and final refitting. The study reports all eight candidate configurations, documents remaining timing concerns and supplies an auditable reproduction package. It does not introduce a new classifier, claim a previously unknown duration problem, or establish an operationally complete pre-call feature set. This restricted scope allows the evidence to be interpreted without overstating novelty or generalizability.

2. Related work and conceptual framework

2.1 Bank telemarketing and benchmark comparability

Moro, Cortez and Rita studied response prediction in a bank telemarketing setting with customer and contextual predictors (Moro, Cortez, & Rita, 2014). Their evaluation used a different data representation and temporal design from the older original-schema dataset used here. That work motivates the application and the need to connect predictive evaluation to campaign decisions; it does not provide an interchangeable numerical baseline. We make no claim of matching or exceeding its performance.

The UCI repository distributes several Bank Marketing variants, including original and additional-feature versions and smaller samples (Moro, Rita, & Cortez, 2014). Results are not directly comparable merely because papers use the same dataset family name. Dataset version, eligible features, class prevalence, split procedure, model configuration and score definition all affect the comparison. Our analysis is confined to the original 45,211-row bank-full.csv schema and to internally matched experimental branches.

2.2 Information availability and reproducibility

Kapoor and Narayanan describe leakage as a broader source of overoptimistic conclusions in machine-learning-based science and emphasize the importance of specifying the relationship between data construction and the intended task (Kapoor & Narayanan, 2023). Here, two safeguards must be distinguished. Fitting preprocessing only on training records protects against one form of cross-partition contamination. Auditing whether an input exists at the prediction timestamp addresses a separate problem. Satisfying the first condition does not establish the second.

Removing duration is consequently treated as one controlled change in the available inputs, not as proof that all temporal leakage has been eliminated. The fields contact, day and month describe the last contact, and campaign includes the last contact in its count. Their suitability depends on whether the operational system supplies historical snapshots, planned-call values or retrospectively recorded observations. The current dataset and experiment do not resolve those alternatives.

2.3 Classifiers and evaluation

The four pipelines cover complementary baseline families: a linear probabilistic model, a bounded decision tree, a randomized tree ensemble and a distance-based classifier. Random forests combine randomized tree predictors and can represent nonlinear interactions without a prespecified linear form (Breiman, 2001). Scikit-learn provides a common implementation framework for these estimators and their preprocessing (Pedregosa et al., 2011). This common interface facilitates inspection of the pipeline, although it does not make the selected configurations equally optimized.

Saito and Rehmsmeier explain why precision-recall evaluation can reveal limitations obscured by accuracy and ROC summaries in imbalanced classification (Saito & Rehmsmeier, 2015). We therefore report both threshold-dependent classification metrics and ranking measures. The project field named PR_AUC is specifically scikit-learn average precision (Scikit-learn developers, n.d.-a). We use the term average precision throughout the manuscript and do not equate it with trapezoidal integration of a precision-recall curve. The literature discussion is focused and contextual rather than a systematic review.

3. Materials and methods

3.1 Data provenance and target

The executed notebook obtained the original-schema data through KaggleHub using the dataset identifier adityamhaske/bank-marketing-dataset, version 3. Its resolver also implements an official UCI download fallback, but that fallback was not the data acquisition path used in the original saved run. The loaded file was bank-full.csv. Provenance is reported at this level because the project used a mirror of the UCI dataset, not a newly collected sample.

The target y was mapped from yes and no to 1 and 0. The original 16 predictors were age, job, marital, education, default, balance, housing, loan, contact, day, month, duration, campaign, pdays, previous and poutcome. The two feature conditions retained all predictors or removed duration alone. Table 1 summarizes the observed dataset and the partitions. The test class counts are recoverable from the archived candidate confusion matrices.

Table 1 Dataset composition and evaluation partitions

Property	Observed value
Records before and after exact deduplication	45,211 / 45,211
Subscribers and non subscribers	5,289 / 39,922
Positive prevalence	11.6985%
Predictors with and without duration	16 / 15

Property	Observed value
Null predictor values reported	0
Candidate fitting records	28,934
Validation records	7,234
Held out test records	9,043
Test subscribers and non subscribers	1,058 / 7,985
Final refit development records	36,168

3.2 Cleaning and preprocessing

The loader standardized column names, validated a binary target, removed exact duplicate records and excluded common identifier or exported-index column names if present. No exact duplicate was removed in this dataset. Numeric infinities were replaced with missing values. The exploratory cell reported no null predictor entries. Categorical unknown values were retained as categories, and the pdays value of -1 was retained as a numeric sentinel rather than being expanded into a separate prior-contact indicator.

Numeric variables were processed using median imputation followed by standardization. Categorical variables were processed using most-frequent imputation and one-hot encoding with unseen categories ignored at transformation time. A ColumnTransformer combined these outputs, and each classifier was attached in a Pipeline. Every learned transformation was fitted on the candidate fitting partition; the final selected pipeline was fitted anew on the development partition. Scaling was shared across pipeline definitions even though tree estimators do not require standardized inputs. No feature selection, resampling or synthetic data generation was performed.

3.3 Shared partitioning and experimental branches

Using random_state=42, a stratified 80/20 split first reserved 9,043 records for testing. A second stratified 80/20 split of the 36,168 development indices reserved 7,234 records for validation, leaving 28,934 for candidate fitting. These correspond approximately to 64%, 16% and 20% of the full dataset. Both feature conditions reused the same index arrays, so changes in candidate performance are not confounded by different held-out customers or different class proportions.

The four candidate pipelines were fitted separately in each feature condition. Validation labels were used only to calculate candidate F1 for selection. All candidate test scores were retained to document the planned comparison. The selected without-duration estimator was identified by the highest validation F1, with model name as a deterministic tie-break rule if required. The selected configuration was then refitted on the complete development partition and evaluated on the same reserved test partition. Neither hyperparameters nor decision thresholds were tuned against the test set.

3.4 Classifier configurations

Table 2 lists the configured parameters. Remaining estimator parameters were left at the installed library defaults. Logistic Regression, Decision Tree and Random Forest used class_weight="balanced", which weights classes inversely to their frequency in the fitting data. KNN instead used inverse-distance neighbor weighting and has no equivalent class_weight argument in this implementation. Accordingly, this is a comparison of specified pipelines rather than an isolated experiment on model family under identical training objectives. The notebook used FAST_MODE=False, corresponding to 160 forest trees.

Table 2 Fixed candidate configurations

Estimator	Configured parameters
Logistic Regression	max_iter=1200; class_weight=balanced; random_state=42
Decision Tree	max_depth=7; min_samples_leaf=15; class_weight=balanced; random_state=42
Random Forest	n_estimators=160; max_depth=13; min_samples_leaf=4; class_weight=balanced; n_jobs=-1; random_state=42
K Nearest Neighbors	n_neighbors=11; weights=distance; n_jobs=-1

These settings were implemented before the reported run and were not selected by a systematic search. The effective behavior of the forest can also change when the number of input features changes because feature subsampling depends on the transformed feature space (Scikit-learn developers, n.d.-b). For KNN, removing duration changes the distance geometry. The ablation therefore evaluates the specified retrained pipelines under two feature conditions, rather than a model-independent causal effect of duration.

3.5 Outcomes and metric definitions

Positive-class precision, recall and F1 refer to subscription, y=1. Accuracy is the proportion of correctly classified test examples. With TP, FP and FN denoting positive-class confusion counts, the threshold-dependent metrics are defined as follows.

Precision = TP / (TP + FP)

Recall = TP / (TP + FN)

F1 = 2TP / (2TP + FP + FN)

Predicted labels were obtained with each fitted estimator's default predict rule. No custom decision threshold was optimized. ROC AUC and average precision were calculated from the positive-class predict_proba scores. ROC AUC summarizes ranking discrimination across false-positive and true-positive trade-offs; average precision weights precision by successive increments in recall (Scikit-learn developers, n.d.-a). Undefined precision or F1 values were assigned zero through zero_division=0.

An always-no predictor was evaluated on the same test set. Its score vector was constant, so ROC AUC was 0.5 and average precision equaled test prevalence, 0.116997. To summarize the duration comparison, the gain for each metric was defined as the with-duration candidate score minus the without-duration score. Absolute gains are reported without significance tests. No customer grouping or temporal cluster adjustment, bootstrap confidence interval or repeated-split analysis was performed.

3.6 Implementation and evidence audit

The standalone notebook contained eight executable cells, including dependency installation, an embedded train.py script, data exploration, candidate training, plotting, ablation reporting, a prediction demonstration and result packaging. All eight completed in Google Colab on a Python 3 Google Compute Engine CPU backend. The exported executed notebook contained no error outputs. Rich-output rendering initially failed in the browser, but the downloaded notebook and archive retained the numerical tables and figures. Later live screenshots displayed the same rounded model scores.

The audit for this manuscript cross-checked model_results.csv, validation_results.csv, duration_ablation_results.csv, baseline.json and model_metadata.json. Derived changes and ratios were computed from the full-precision stored values. No new classifier training is represented as part of this manuscript preparation. The accompanying package includes the executed notebook, extracted training script, tables, archived figures and a file-hash manifest. Exact installed package versions and a raw-data content hash were not captured during the original execution.

4. Results

4.1 Candidate classification performance

Table 3 reports all eight candidate evaluations. Random Forest had the largest F1 in both feature conditions: 0.5862 with duration and 0.4569 without duration. It also had the largest ROC AUC and average precision in both conditions. With duration, Decision Tree attained the highest recall, 0.8544, whereas KNN attained the highest accuracy, 0.8987, and precision, 0.6355. Without duration, Logistic Regression had the highest recall, 0.6304; KNN again had the highest accuracy and precision, but its recall was only 0.1739.

Table 3 Candidate test performance on identical held out records

Duration	Model	Accuracy	Precision	Recall	F1	ROC AUC	AP
With	Logistic Regression	0.8445	0.4162	0.8166	0.5514	0.9081	0.5373
With	Decision Tree	0.7904	0.3418	0.8544	0.4883	0.8867	0.5091
With	Random Forest	0.8628	0.4529	0.8308	0.5862	0.9240	0.6040
With	KNN	0.8987	0.6355	0.3147	0.4210	0.8717	0.5134
Without	Logistic Regression	0.7551	0.2678	0.6304	0.3759	0.7720	0.4091
Without	Decision Tree	0.8353	0.3659	0.5558	0.4413	0.7562	0.3524
Without	Random Forest	0.8391	0.3775	0.5784	0.4569	0.7971	0.4430
Without	KNN	0.8893	0.5916	0.1739	0.2688	0.7284	0.3591

AP denotes average precision. All rows in Table 3 were fitted on 28,934 records and evaluated on 9,043 records. The models were not selected by these test values. Figure 1 shows the F1 comparison; ranking metrics are reported in Table 3. The contrast between KNN accuracy and F1 is particularly

pronounced without duration: accuracy is 0.8893 while F1 is 0.2688. The always-no baseline attains accuracy 0.8830, so high accuracy in this task is not itself evidence of effective subscriber detection.

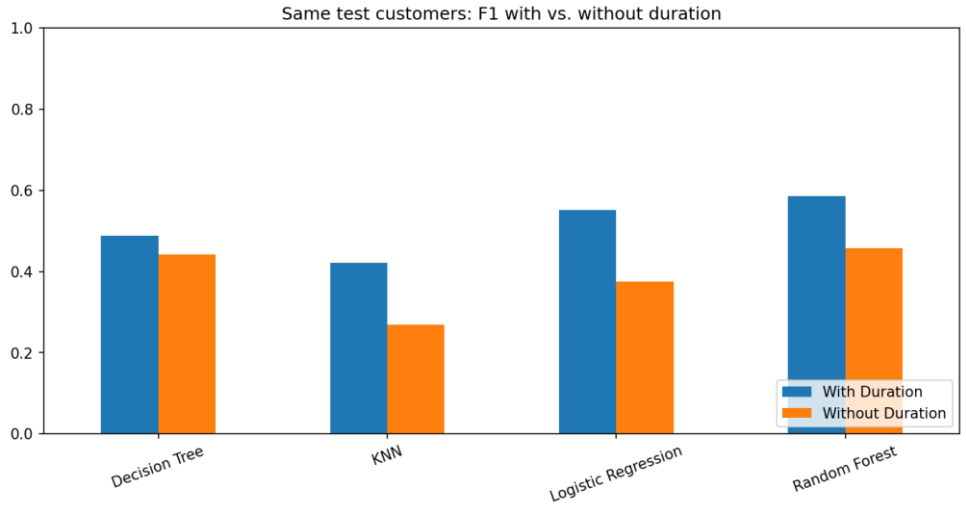


Figure 1 Candidate F1 with and without duration. This is the original plot exported by the Colab training pipeline. Every bar refers to the same test partition; the final refitted model is reported separately in Table 5.

4.2 Magnitude of the duration association

Removing duration reduced F1, ROC AUC, average precision and recall for every implemented classifier. Table 4 reports the absolute gains associated with including duration. Random Forest lost 0.1293 F1, 0.1269 ROC AUC and 0.1610 average precision when duration was removed. Relative to its with-duration candidate F1, the reduction was approximately 22.06%. KNN showed the largest relative F1 reduction, approximately 36.15%, whereas Decision Tree showed the smallest, approximately 9.62%.

Table 4 Absolute candidate metric gains from including duration

Model	F1 gain	ROC AUC gain	AP gain	Recall gain
Logistic Regression	0.1755	0.1360	0.1283	0.1862
Decision Tree	0.0470	0.1306	0.1567	0.2987
Random Forest	0.1293	0.1269	0.1610	0.2524
KNN	0.1522	0.1433	0.1543	0.1408

The effect was not uniform across metrics. Decision Tree accuracy increased from 0.7904 with duration to 0.8353 without duration, even as its F1 decreased from 0.4883 to 0.4413. KNN precision decreased from 0.6355 to 0.5916, while its low positive prediction rate remained evident in the confusion counts. More generally, the absolute gain depends on the classifier, its training objective and its decision rule. These differences are descriptive observations from one matched split; they do not establish statistical superiority or causal effects on customer behavior.

4.3 Validation selection and final refit

The without-duration validation F1 values were 0.3764 for Logistic Regression, 0.4483 for Decision Tree, 0.4607 for Random Forest and 0.2870 for KNN. Random Forest was therefore selected by the prespecified rule. Its validation advantage over Decision Tree was approximately 0.0124 F1, a small difference for which no uncertainty interval was estimated. The selected forest was refitted on all 36,168 development examples. Table 5 separates this final pipeline from the candidate fit and from the always-no baseline.

Table 5 Without duration forest before and after final refitting

Metric	Candidate RF	Final refit RF	Always no
Accuracy	0.8391	0.8362	0.8830
Precision	0.3775	0.3725	0.0000

Metric	Candidate RF	Final refit RF	Always no
Recall	0.5784	0.5841	0.0000
F1	0.4569	0.4549	0.0000
ROC AUC	0.7971	0.8001	0.5000
Average precision	0.4430	0.4470	0.1170

The final saved model achieved accuracy 0.8362, precision 0.3725, recall 0.5841, F1 0.4549, ROC AUC 0.8001 and average precision 0.4470. Relative to the candidate forest, the refit increased recall by approximately 0.0057 and ROC AUC by 0.0030, while F1 decreased by approximately 0.0020. Additional fitting data therefore did not improve every metric on this test sample. These small changes are not interpreted as evidence of a meaningful improvement or deterioration.

4.4 Error trade offs

The without-duration candidate forest identified 612 of the 1,058 subscribers and missed 446, while producing 1,009 false positives among 7,985 non-subscribers. KNN identified only 184 subscribers and missed 874, but produced just 127 false positives. Logistic Regression identified 667 subscribers and produced 1,824 false positives. Decision Tree occupied an intermediate operating point, with 588 true positives and 1,019 false positives. These counts explain why the classifiers rank differently under precision, recall and accuracy.

For completeness, the final-refit confusion counts can be reconstructed uniquely from the saved full-precision precision, recall and accuracy together with the test class totals: TP=618, FN=440, FP=1,041 and TN=6,944. This reconstruction is an algebraic audit of the stored metrics, not a newly exported prediction-level evaluation. It yields 7,562 correct labels among 9,043 test records and 1,659 predicted positive labels. Table S1 in the supplement documents the reconstruction and candidate counts.

4.5 Ranking quality and feature attribution

The archived without-duration candidate ROC and precision-recall curves are provided as Figures S1 and S2. Random Forest achieved candidate ROC AUC 0.7971 and average precision 0.4430; the final-refit values were 0.8001 and 0.4470. The final average precision was approximately 3.82 times the constant-score baseline AP. This numerical ratio compares AP values; it is not measured lift at a contact-budget threshold and should not be interpreted as an increase in campaign profit.

The candidate forest's largest individual transformed-feature importances were balance (0.0885), age (0.0858), poutcome_success (0.0823) and day (0.0720). The archived plot is provided in Figure S4. These are impurity-based importances, and categorical fields are represented across multiple encoded levels. The values neither establish causal drivers nor remove the need to audit timing. In particular, prominence of a last-contact field does not make that field available before contact.

5. Discussion

5.1 Predictive performance depends on the decision timestamp

The matched comparison shows a consistent association between inclusion of duration and higher F1, ROC AUC and average precision for the implemented configurations. The result supports the practical importance of declaring the prediction timestamp. A retrospective score using a completed call's duration describes a different information set from one available before that call. The distinction remains important even when both scores are numerically valid and measured on records withheld from model fitting.

This finding should not be presented as a new discovery of duration leakage. Its value is the transparent quantification of the change under one reproducible protocol, together with an explicit description of the unresolved information constraints. There is also no claim that duration alone determines response. The retrained pipelines may use its information differently, and removal changes the representation seen by each learner. An explanation of the observed behavior must therefore remain conditional on the specified preprocessing and model settings.

5.2 Accuracy and the subscriber detection objective

The always-no baseline illustrates why a strong overall accuracy can coexist with no useful positive-class detection. KNN without duration improves accuracy only modestly over this baseline while missing more than four fifths of the subscribers. Random Forest sacrifices some overall accuracy in exchange for substantially higher recall and F1. Neither trade-off is universally preferable: the appropriate decision depends on calling capacity, contact cost, customer burden and the value of the desired outcome.

Class weighting contributes to these differences. The three weighted estimators and the unweighted-class KNN do not optimize identical empirical objectives. Thus the observed ranking supports selection among the implemented pipelines under validation F1, not a general claim that forests dominate nearest-neighbor methods. A fairer algorithm-family benchmark would compare validated hyperparameter ranges and explicitly aligned sampling, weighting and threshold policies. Such a benchmark was outside the completed experiment.

5.3 Implications for campaign evaluation

The final model's precision of 0.3725 means that approximately 37 of every 100 flagged records were subscribers in this test set. Recall of 0.5841 means that approximately 58 of every 100 actual subscribers were detected. These interpretations are useful for communicating the observed operating point, but they are not an estimate of incremental response caused by contacting those customers. Observational prediction does not identify which customers would subscribe without a call or which customers are persuaded by one.

A practical next evaluation would define the allowed information at contact selection time, rank customers using development-selected policies and report precision, recall and lift at explicit calling budgets. Calibration would be needed before scores are used as probability estimates in cost calculations. Prospective or external validation would then assess whether the policy transfers to a later campaign. The current experiment supplies an inspectable baseline for those steps rather than evidence that they have already succeeded.

5.4 Selection discipline after results become visible

The notebook separates selection from test scoring, and the manuscript preserves that distinction. However, publishing or inspecting test results consumes their role as an unseen evaluation resource for subsequent model development. Future choices of features, estimators or thresholds should not be justified by repeated improvement on this same visible test partition. A new evaluation protocol should be fixed before testing and should, where possible, use a later period or an independent dataset. Reusing the present split solely to document reproducibility is different from tuning against it.

6. Limitations

The first limitation is incomplete feature-timing validation. Duration was removed, but contact, day, month and campaign remained. Some historical customer and previous-campaign variables may be usable before a call only if their values are frozen at the relevant timestamp. No strict pre-call reconstruction was performed. The label "without duration" is therefore the primary description of the condition; "pre-call candidate" is a prospective use hypothesis, not an established property of the model.

Second, the evaluation uses one stratified random split of historical records associated with a single bank. It does not establish performance on future campaigns, another institution, another population or a changed economic environment. Exact duplicate removal is insufficient to prove customer independence. Because a grouping audit was not performed, dependence between repeated observations of a customer could remain across partitions. A chronological or customer-grouped protocol may yield different estimates.

Third, no confidence intervals, significance tests, repeated seeds or repeated resampling results were computed. In particular, the small validation F1 margin between Random Forest and Decision Tree cannot be interpreted as a statistically established advantage. ROC AUC and average precision uncertainty cannot be recovered from aggregate scores alone without additional assumptions or prediction-level data. We therefore avoid significance claims and do not present constructed uncertainty bands.

Fourth, the model space is intentionally restricted. Fixed hyperparameters, dense one-hot encoding, numeric treatment of the pdays sentinel, differing class-weight policies and default decision rules all affect performance. Stronger tabular baselines, structured feature engineering, calibration and development-only threshold tuning were not assessed. The observed best model is the winner within this limited implemented comparison, not the best attainable classifier for the application.

Fifth, the run is only partially specified for exact computational replication. Random seeds, source code, result tables and model metadata are preserved, but the original installation used dependency ranges rather than a frozen package manifest. Unspecified defaults can vary across library releases. The raw-data file was not archived with a checksum. The supplied hash manifest verifies the preserved artifacts; it cannot retrospectively establish the byte-level identity of the original data or the exact installed environment.

Finally, no prospective marketing outcome, causal uplift, profitability, calibration or fairness analysis was conducted. Numeric account attributes and demographic fields require appropriate governance in any operational use. The project's UI convenience defaults and category lists were derived from the full feature frame, although those values did not determine the reported fitted transformations or predictions. A deployment specification should replace them with development-only or independently approved input rules.

7. Conclusion

On the original Bank Marketing dataset, all four implemented classifiers showed lower F1, ROC AUC, average precision and recall when call duration was removed. Random Forest had the highest candidate F1 in both conditions, with a reduction from 0.5862 to 0.4569. Validation F1 selected the without-

duration forest, whose final refit attained F1 0.4549, ROC AUC 0.8001 and average precision 0.4470 on 9,043 held-out records. KNN's higher accuracy was accompanied by markedly lower subscriber recall.

The study demonstrates the need to report the available information set, the selection metric and the distinction between candidate and final-refit evaluations. Its contribution is an auditable applied baseline rather than a novel learning method or a validated deployment policy. A strict feature-availability audit, independent temporal evaluation and decision-relevant contact-budget analysis are the next requirements for stronger operational and publication claims.

Data and code availability

The source data are available through the UCI Bank Marketing repository (Moro, Rita, & Cortez, 2014); the completed run used the stated Kaggle mirror. Supplementary Material S1 supplies the executed notebook, embedded training script, full-precision result tables, model metadata, baseline scores, archived plots, model artifact and integrity manifest. The package does not include the raw dataset. No permanent public code repository or archive DOI has yet been assigned. The original Colab notebook is linked in the supplement; access depends on its existing sharing settings.

Transparency statement

Generative AI assistance was used for manuscript drafting, literature retrieval, numerical auditing, figure preparation and document production. All three named authors have approved submission to IJRPR and remain responsible for the manuscript content.

Declarations

Funding: This study received no funding.

Competing interests: The authors declare no competing interests.

Author contributions: Arjun Ananthakrishnan and S Sreya contributed equally to this study. Dr. G. Paavai Anand provided faculty supervision and advisory guidance.

Ethics statement: This study used publicly available secondary data. No institutional ethics approval or exemption was obtained.

Submission declaration: The authors confirm that this manuscript is unpublished, is not under consideration elsewhere, and has been approved for submission by all three authors.

References

Breiman, L. (2001). Random forests. Machine Learning, 45, 5–32. https://doi.org/10.1023/A:1010933404324

Kapoor, S., & Narayanan, A. (2023). Leakage and the reproducibility crisis in machine-learning-based science. Patterns, 4(9), Article 100804. https://doi.org/10.1016/j.patter.2023.100804

Moro, S., Cortez, P., & Rita, P. (2014). A data-driven approach to predict the success of bank telemarketing. Decision Support Systems, 62, 22–31. https://doi.org/10.1016/j.dss.2014.03.001

Moro, S., Rita, P., & Cortez, P. (2014). Bank Marketing [Data set]. UCI Machine Learning Repository. https://doi.org/10.24432/C5K306

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, É. (2011). Scikit-learn: Machine learning in Python. Journal of Machine Learning Research, 12, 2825–2830. https://jmlr.org/papers/v12/pedregosa11a.html

Saito, T., & Rehmsmeier, M. (2015). The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets. PLOS ONE, 10(3), Article e0118432. https://doi.org/10.1371/journal.pone.0118432

Scikit-learn developers. (n.d.-a). average_precision_score. Retrieved October 30, 2026, from https://scikit-learn.org/stable/modules/generated/sklearn.metrics.average_precision_score.html

Scikit-learn developers. (n.d.-b). RandomForestClassifier. Retrieved October 30, 2026, from https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html