Predicting Portuguese Bank Marketing Outcomes with Support Vector Machines: A Comparative Analysis

Naomi Buell[1] and Richie Rivera[1]

[1]CUNY School of Professional Studies

November 8, 2025

1 Introduction

We analyzed a bank marketing dataset that records the outcomes of a Portuguese bank's telemarketing campaign for term deposits (Sérgio Moro, Paulo Cortez, and Paulo Rita 2014). Our goal was to apply Support Vector Machine (SVM) and three tree-based machine learning techniques to compare algorithms and choose the optimal model.

2 Literature Review

As background for this assignment, we first reviewed the two assigned articles on predicting COVID-19 cases (Ahmad et al. 2021; Guhathakurata et al. 2021). The Ahmad (2021) article compared various decision tree ensembles, while the Guhathakurata (2021) article compared KNN, Naïve Bayes, Random Forest, AdaBoost, a binary tree, and SVM. Guhathakurata found that SVMs outperformed other models, whereas Ahmad found that decision tree ensembles performed best. We will generate the same metrics used in these papers (accuracy, recall, precision, and F1-score) for this study.

We also reviewed three additional articles comparing decision trees and SVMs in the authors' areas of expertise. A 2025 study on predicting chronic kidney disease compared Random Forest, SVM, Naive Bayes, Logistic Regression, KNN, and XGBoost (Phuong et al. 2025). It found that Random Forest, XGBoost, SVM, and logistic regression performed best, each achieving 100% accuracy. A 2021 study by used six ML models (logistic regression, SVM, kNN, random forest, neural network, and AdaBoost) to predict patient dropout from alcohol use disorder (AUD) treatment (Park et al. 2021). The AdaBoost model was determined to be the best. A 2022 study formally compared Random Forest, SVM, and LSTM for predictive maintenance on aircraft engines (Azyus and Wijaya 2022). For classification, LSTM performed best (98.7% accuracy), while for regression, Random Forest had the lowest RMSE.

This review highlights that the optimal algorithm is highly dependent on the specific application, with SVMs, Decision Trees, and ensemble methods like Random Forest and AdaBoost all proving to be the best performing choice in different contexts.

3 Methods

3.1 Pre-processing

We used the full bank marketing dataset (41,188 records, 21 attributes) containing client-level and macroeconomic variables from May 2008 to November 2010 (S. Moro, P. Rita, and P. Cortez 2014). We handled missingness by removing columns with substantial gaps or low predictive value (`default`, `pdays`) and dropping rows with minor missingness. We excluded the `duration` variable due to data leakage concerns and removed 1,946 duplicate records (5.1%) to improve data quality.

Feature encoding was critical. Ordinal features (`education`, `month`, `day_of_week`) were manually mapped to their numerical order. Nominal features (e.g., `job`, `marital`) were one-hot encoded. Because SVMs are not scale-invariant, all numerical features were standardized using the `RobustScaler` package; we chose this package specifically to reduce the influence of outliers present in the data. For all experiments, we split the data into 30% testing and 70% training sets.

3.2 Models

For this assignment, we tried four new support vector machine configurations (1-4 below). We also recap the models in the previous assignment (5-10 below):

1. SVM (Linear): an SVM with a linear kernel with the regularization parameter C=1.0.
2. SVM (RBF): an SVM the same as #1 above but with an RBF (radial basis function) kernel. The RBF kernel is a popular choice that allows SVMs to solve non-linear problems by projecting data into a higher-dimensional space.
3. SVM (GridSearch): an SVM the same as #2 above but uses a grid search to find the optimal hyper-parameters, which yielded C=10 and gamma set to 'scale'.
4. SVM (GridSearch, Undersampled): an SVM the same as #3 above but fit to balanced (undersampled) data. This was done to address the severe class imbalance, a common problem in real-world datasets. This model yielded optimal hyper-parameters C= 1 and gamma = 0.001. **We determined that this was our optimal SVM model** because it substantially improved recall compared to the other SVM models above. This means the model is able to identify a much larger proportion of the positive class (subscribed) instances.
5. Decision tree (Depth 4): a decision tree with a max depth of 4
6. Decision tree (Depth 2): a decision tree with a max depth of 2
7. Random Forest (n=25): a random forest with 25 trees
8. Random Forest (GridSearch): a random forest optimized with a grid search
9. AdaBoost (n=50, lr=1): an AdaBoost model with 50 estimators and a learning rate of 1.0
10. AdaBoost (GridSearch, Undersampled): an AdaBoost model using an undersampled dataset and optimized with a grid search

In the next section, we will compare our optimal SVM model (grid-search, under-sampled) with the six approaches from our previous assignment.

4 Results

The results in Table 1 highlight a clear trade-off between accuracy and recall, driven by the dataset's class imbalance. The eight models trained on the original, imbalanced data achieved high test accuracy (89.3%-90.0%), but all suffered from poor positive-class recall (17.3%-29.7%). This indicates they excelled at predicting the majority class (non-subscription) but largely failed to identify the minority class (subscription).

In contrast, the two models trained using undersampled data—SVM (GridSearch, Undersampled) and AdaBoost (GridSearch, Undersampled)—showed a significant increase in positive-class recall (69.8%-70.5%). It's expected that this improvement would come at the cost of lower overall accuracy (fell to approximately 72%-73.3%). The AdaBoost (GridSearch, Undersampled) model achieved the highest macro recall (71.7%), suggesting it provided the most balanced performance across both classes.

Model	Accuracy	Recall (positive class)	Precision (macro)	Recall (macro)	F1 (macro)	Optimal parameters
SVM (Linear)	0.898	0.193	0.771	0.589	0.620	N/A
SVM (RBF)	0.898	0.213	0.767	0.599	0.631	N/A
SVM (GridSearch)	0.897	0.241	0.754	0.610	0.644	C: 10; gamma: scale
SVM (GridSearch, Undersampled)	0.720	**0.705**	0.596	0.713	0.590	C: 1; gamma: 0.001
Decision Tree (Depth 4)	0.900	0.177	0.797	0.584	0.614	N/A
Decision Tree (Depth 2)	0.900	0.173	0.801	0.582	0.612	N/A
Random Forest (n=25)	0.893	0.297	0.727	0.632	**0.662**	N/A
Random Forest (GridSearch)	0.893	0.297	0.727	0.632	**0.662**	Max depth: None; n estimators: 25
AdaBoost (n=50, lr=1)	**0.900**	0.174	**0.802**	0.582	0.612	N/A
AdaBoost (GridSearch, Undersampled)	0.733	0.698	0.600	**0.717**	0.599	Learning rate: 0.85; n estimators: 20

Table 1: Summary of Results by Experiment. *Best-performing models in each column are shown in bold.*

5 Conclusion

To get the most accurate results, the AdaBoost model with 50 estimators and a learning rate of 1 performed the best, accurately predicting 90% of test cases.

However, we'd recommend prioritizing recall for this use case because the bank likely wants to identify as many potential subscribers as possible (even if that means calling some customers who end up being false positives). With the priority of increasing recall, we recommend using an SVM with hyperparameter tuning and data balancing. This model provided the best recall performance for predicting the positive class (subscribers) relative to tree-based models and correctly identified over 70% of actual subscribers. However, this came at the cost of low precision and many false positives. If the bank is okay with this tradeoff, then this model can be used in production.

Alternatively, if the business goal is to gain insights into understanding which factors would result in a subscriber, the AdaBoost model with a learning rate of 0.85 and 20 estimators offers strong interpretability and predictive power over the SVM model with hyperparameter tuning and data balancing. It achieved the highest macro recall out of all models (71.7%), indicating that the model performs well across predicting both subscriber and non-subscriber classes, rather than being biased toward the majority class (non-subscribers). However, this high recall comes at the cost of lower precision (macro precision = 60%), meaning on average, across both subscribers and non-subscribers, when the model predicts a class, it's correct about 60% of the time. This is a reasonable trade-off as we are finding a relatively rare class (subscribers) within the total population.

Both the SVM and AdaBoost models are better suited for classification scenarios rather than regression scenarios. This makes sense as both algorithms are designed to predict discrete class labels (e.g., subscriber or not) rather than continuous outcomes. Overall, we recommend the SVM with data balancing if the goal is to maximize subscriber identification, or the tuned AdaBoost model if interpretability and balanced performance are prioritized.

References

Ahmad, Amir et al. (2021). “Decision Tree Ensembles to Predict Coronavirus Disease 2019 Infection: A Comparative Study”. In: *Complexity* 2021, p. 5550344. DOI: 10.1155/2021/5550344. URL: https://www.hindawi.com/journals/complexity/2021/5550344/.

Azyus, Fitra and A. Wijaya (2022). “Determining the Method of Predictive Maintenance for Aircraft Engine Using Machine Learning”. In: *Journal of Computer Science and Technology Studies* 4 (1), pp. 1–11. DOI: 10.32996/JCSTS.2022.4.1.1.

Guhathakurata, S. et al. (2021). “A novel approach to predict COVID-19 using support vector machine”. In: *Data Science for COVID-19*. Academic Press, pp. 351–364. DOI: 10.1016/B978-0-12-824536-1.00014-9. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8137961/.

Moro, S., P. Rita, and P. Cortez (2014). *Bank Marketing*. UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C5K306.

Moro, Sérgio, Paulo Cortez, and Paulo Rita (2014). “A data-driven approach to predict the success of bank telemarketing”. In: *Decision Support Systems* 62, pp. 22–31.

Park, Sun Jae et al. (2021). “Machine learning prediction of dropping out of outpatients with alcohol use disorders”. In: *PLOS ONE* 16 (8), e0255626. DOI: 10.1371/journal.pone.0255626. URL: https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0255626.

Phuong, Nguyen Dinh et al. (2025). “Machine Learning Techniques in Chronic Kidney Diseases: A Comparative Study of Classification Model Performance”. In: *Bioinformatics and Biology Insights* 19, p. 11779322251356563. DOI: 10.1177/11779322251356563. URL: https://pubmed.ncbi.nlm.nih.gov/40735336/.