Machine Learning for Bank Marketing Campaigns

GROUP MEMBER LIST

Name	Matriculation Number
CHENG YUTING	A0303704R
PAN SAI	A0304252U
WANG RUOXI	A0304225U
WANG ZIRONG	A0303891A

13/04/2025

Section 1 Introduction

1. Research Motivation and Significance

In the field of machine learning, predictive modelling has become a crucial tool for various applications, including finance, healthcare, and marketing. Accurate prediction of customer behaviour not only optimizes marketing resources but also enhances customer satisfaction by personalizing offers based on user preferences. This study aims to explore and compare the performance of different machine learning models, particularly methods such as Random Forest and GBM, in predicting a binary outcome using a comprehensive dataset from the banking industry. Previous research has shown that these ensemble methods are effective in handling complex datasets with high dimensionality and mixed feature types (Breiman, 2001; Breiman, 1996). This study contributes to the existing literature by applying and comparing these methods on a real-world dataset, providing valuable insights into their performance and applicability.

2. Literature Review

Several studies have explored the use of machine learning algorithms in predicting customer behaviour in banking. For instance, Lessmann et al. (2015) compared various machine learning techniques, including Random Forest and Bagging, and highlighted their effectiveness in handling imbalanced datasets and complex interactions between features. Similarly, Lessmann et al. (2016) demonstrated that ensemble methods outperform traditional statistical models in terms of prediction accuracy and robustness. More recently, studies have emphasized the importance of feature engineering and hyperparameter tuning in enhancing the performance of these models (Kuhn, 2008).

3. Main Findings

This study evaluates various machine learning models and ensemble techniques on a banking dataset. The results indicate that ensemble models consistently outperform individual models such as logistic regression, decision trees, and random forests. Notably, the GR unrestricted model (excluding the constant term) achieves the highest AUC scores, demonstrating the effectiveness of integrating diverse models to improve predictive performance.

4. Structure of the Report

The report is organized as follows: Section 2 details the data and preprocessing. Section 3 covers methodology and implementation using R packages. Section 4 presents the results with performance metrics. Section 5 discusses the findings. Section 6 concludes with key findings and future research recommendations.

Section 2 Data

1. Description of the dataset

- The data comes from direct marketing campaigns conducted by a Portuguese banking institution, which were carried out through phone calls. The goal is to predict whether a client will subscribe to a term deposit (variable *y*). In many cases, multiple calls to the same client were needed to determine whether they would eventually subscribe ('yes') or not ('no') to the product.
- The data consists of 41,188 rows, 20 inputs and one output, of which the 20 inputs can be subdivided into four categories: Personal Attributes, Campaign Contact Attributes, Interaction History, and Economic Indicators.

2. Data Preprocessing

- Data Description and Feature Engineering:
  - Descriptive statistics show that we have 10 quantitative variables and 11 qualitative variables out of a total of 21 (*y* is among them).
  - Upon closer inspection, we noticed something unusual about the *pdays* variable. This variable represents the number of days that passed by after the client was last contacted from a previous campaign, where almost the majority of the values are 999 (999 means client was not previously contacted), with the rest of the values distributed between 6 and 20. Obviously 999 is not the true number of days since the client was last contacted, it is a signal that the client has not yet been contacted, then the numerical level is meaningless, we need to use the features to transform it into a 0-1 variable (0 means not yet contacted, 1 means contacted in the last month)
  - After adjustments we have a total of 9 quantitative variables and 12 qualitative variables.
- Data Cleaning:
  - We observe that there is a partial UNKNOWN value in the qualitative variable, when the value is UNKNOWN, it does not represent any categorical category, it is just not categorized, such a value does not give us any information and does not help us in the subsequent categorization prediction, so we consider removing all the rows that contain UNKNOWN, and we are left with the final rows of 30488.

3. Data Segmentation

We divide all the data into training set (60%), validation set (20%), and test set (20%) in the ratio of 3:1:1. To address the class imbalance—since the positive class (y = 1) is relatively rare—we apply stratified sampling to ensure that the proportion of positive cases remains consistent across all three subsets. This approach helps prevent underfitting in the test set due to insufficient data while maintaining enough data for parameter estimation and model validation.

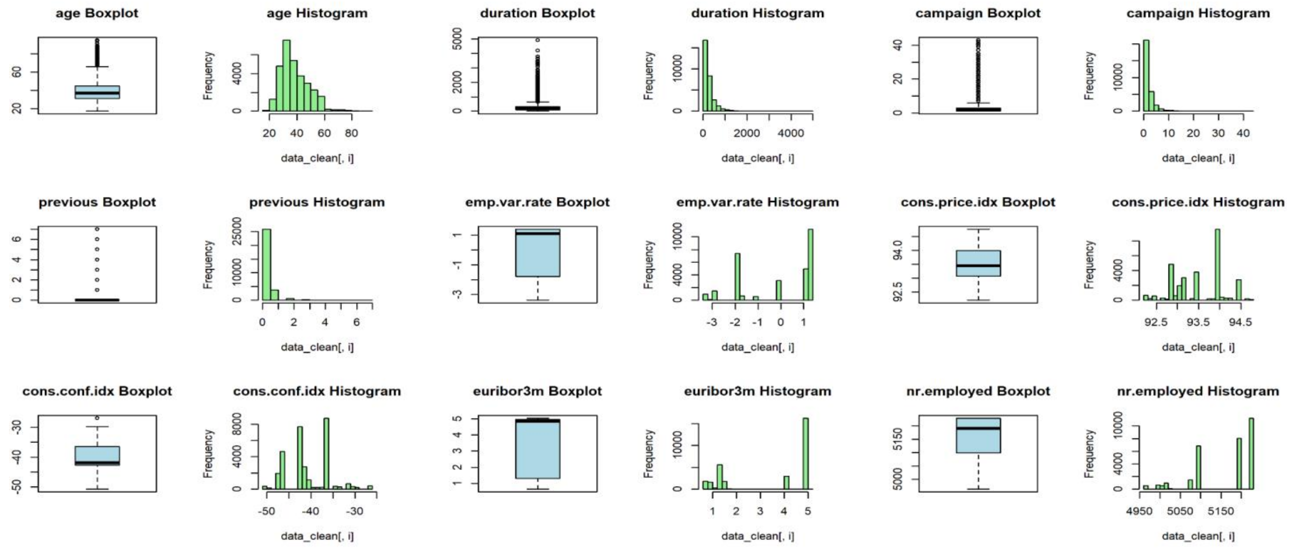

Figure 1 Visualization-9 Quantitative Variables

Table 1 Overview of Variable Distributions

Variable	Definition	Boxplot	Histogram	Overall Characteristics
Age	Client's age (typically adult)	Concentrated around 30–40, few outliers	Right-skewed, peak ~mid-30s	Young to middle-aged dominant; few older/younger outliers
Duration	Duration of last contact (in seconds)	Small IQR, many large outliers	Strong right skew, many short calls, few very long ones	Majority short calls; long ones are rare but significant
Campaign	Number of contacts in current campaign	Small IQR, upper outliers	Right-skewed, most contacted 1–2 times	Most clients contacted once/twice; few contacted many times
Previous	Contacts in previous campaigns	Mostly zero, few higher values	Zero-inflated, nonzero right-skewed	Majority never contacted before; small group contacted multiple times
Emp.var.rate	Employment variation (quarterly)	Narrow IQR, some extreme outliers	Slightly skewed or unimodal	Moderate negative to small positive; reflects changing economy
Cons.price.idx	Consumer Price Index (monthly)	Tight distribution, minimal outliers	Concentrated around peak (e.g., 93–94)	Stable within small range; low variation
Cons.conf.idx	Consumer Confidence Index (monthly, negative)	Compact IQR, few extreme outliers	Unimodal or slightly skewed	Mostly negative, narrow range with few extreme months
Euribor3m	3-month Euribor rate (daily)	Wide IQR if rates fluctuate; outliers possible	Unimodal or multimodal	Varies with time; multiple peaks for different rate regimes
Nr.employed	Number of employees (quarterly, in thousands)	Narrow IQR, minimal outliers	Highly concentrated range (e.g., 5000–5200)	Very stable over time; few extreme values

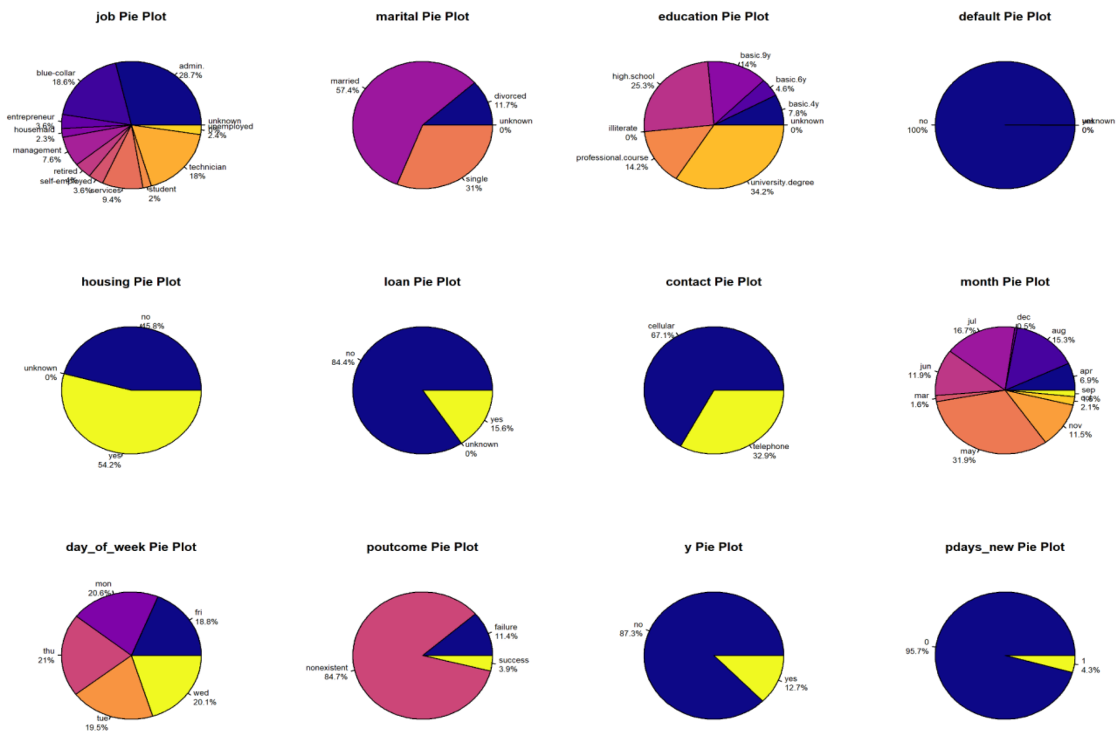

Figure 2 Visualization-12 Qualitative Variables

Table 2 Key Insights from Client Demographics and Marketing Outcomes

Variable	Main Finding	Analysis
Job	Admin. (28.7%), blue-collar (18.6%), technician (16.8%)	These three occupations dominate the dataset, indicating that the client base primarily comprises administrative, technical, and manual labor sectors.
Marital Status	Married clients account for 57.4%	The majority being married suggests prevailing societal marital norms and possibly household-level decision-making patterns.
Education	University degree (34.2%) and high school (25.3%)	The dataset reflects a relatively well-educated clientele, with most having completed at least secondary education.
Default	100% of clients have no default records	This implies a consistently strong credit history across all clients in the dataset.
Housing Loan	Yes (54.2%), No (45.8%)	Housing loan distribution is balanced, potentially reflecting regional mortgage accessibility and economic stability.
Personal Loan	84.4% of clients have no personal loan	A low rate of personal loans may indicate financial conservatism or limited access to unsecured credit.
Contact Method	Cellular (67.1%)	The high usage of mobile phones for communication reflects modern contact preferences and widespread mobile adoption.
Month	May (31.9%), July (16.7%), August (15.3%), June (11.9%)	Marketing efforts were concentrated in late spring and summer months, possibly aligning with seasonal campaigns or strategic outreach periods.

Day of Week	Thursday (21%), Monday (20.6%), Wednesday (20.1%)	Contact attempts are concentrated mid-week, possibly to maximize client availability and response rates.
Previous Outcome	84.7% marked as “nonexistent”	The majority of clients had no involvement in past campaigns, indicating limited prior engagement.
Subscription (y)	87.3% of clients did not subscribe to term deposits	The low conversion rate may point to suboptimal campaign performance or client disinterest.
Pdays_new	95.7% of clients were not recontacted	This suggests minimal follow-up activity, which may have contributed to the low subscription rate.

Section 3 Methodology

This study employs a systematic machine learning approach to predict a binary outcome using a banking dataset. The methodology involves comprehensive data preprocessing, including cleaning and feature engineering, to ensure data quality. A diverse set of algorithms, such as logistic regression, Lasso regression, Ridge regression, Elastic Net, decision trees, random forests, GBM and neural networks, were selected for their ability to handle varying complexities and data characteristics. Each model underwent parameter tuning using techniques like grid search and cross-validation to optimize performance. Model evaluation was conducted using key metrics including AUC, MSE and S.E. to ensure a thorough assessment of predictive power and reliability. This structured approach ensures a robust comparison of model performance and applicability to the dataset.

1.Logistic Regression

1.1 Model Description

Logistic regression was chosen as a baseline model due to its interpretability and effectiveness in binary classification tasks, providing a clear benchmark for comparing more complex algorithms.

1.2 Implementation Details

- **Model Training**: using the glm function from the stats package in R, with family = ＂binomial＂ to handle binary outcomes. This function estimates probabilities based on predictor variables and fits the model to the training data.
- **Threshold Optimization:** the optimal classification threshold was determined using the ROC curve computed by the pROC package, identifying the point that maximizes the balance between sensitivity and specificity.

Achieving 0.9340 AUC, 0.0659 MSE, and 0.0022 S.E.

2. Lasso Regression

2.1 Model Description

Lasso regression, a regularization technique, was chosen to address multicollinearity and feature selection in high-dimensional datasets. By adding an L1 penalty to the regression coefficients, Lasso shrinks less important feature coefficients to zero, effectively performing feature selection and improving model interpretability.

2.2 Implementation Details

- **Parameter Grid Setup**: Defined a parameter grid lasso_grid with a fixed alpha = 1 (indicating Lasso regression) and different lambda values (10^seq(-4, 0, length = 50)).
- **Model Training**: Used the train function with the glmnet method. Employed 5-fold cross-validation (trainControl) to search for the optimal lambda value in the parameter grid. The training data was trainData, and the optimization metric was the area under the ROC curve (metric = "ROC").

Achieving 0.9341 AUC, 0.0658 MSE, and 0.0022 S.E.

3. Ridge Regression

3.1 Model Description

Ridge regression, a regularization technique, was chosen to address multicollinearity in datasets. By adding an L2 penalty to the regression coefficients, Ridge shrinks the coefficients of less important features, reducing model complexity and overfitting while maintaining all features in the model.

3.2 Implementation Details

- **Parameter Grid Setup**: Defined a parameter grid ridge_grid with a fixed alpha = 0 (indicating Ridge regression) and different lambda values (10^seq(-4, 0, length = 50)).
- **Model Training**: Used the train function with the glmnet method. Employed 5-fold cross-validation (trainControl) to search for the optimal lambda value in the parameter grid. The training data was trainData, and the optimization metric was the area under the ROC curve (metric = "ROC").

Achieving 0.9313 AUC, 0.0673 MSE, and 0.0022 S.E.

4. Revised Elastic Net

4.1 Model Description

Elastic Net regression, a hybrid regularization technique, was chosen to combine the strengths of both Lasso (L1) and Ridge (L2) regression. By adding a mix of L1 and L2 penalties to the regression coefficients, Elastic Net effectively performs feature selection while handling multicollinearity, making it suitable for high-dimensional datasets with correlated features.

4.2 Implementation Details

- **Parameter Grid Setup**: Defined a parameter grid en_grid with alpha values ranging from 0 to 1 (in increments of 0.2) and different lambda values (10^seq(-4, 0, length.out = 20)).
- **Model Training**: Used the train function with the glmnet method. Employed 5-fold cross-validation (trainControl) to search for the optimal combination of alpha and lambda values in the parameter grid. The training data was trainData, and the optimization metric was the area under the ROC curve (metric = " ROC ").

Achieving 0.9341 AUC, 0.0658 MSE, and 0.0022 S.E. (The same as Lasso)

5. Decision Tree

5.1 Model Description

We implemented a CART decision tree using the rpart package to capture non-linear relationships and feature interactions. The model provides explicit feature importance rankings while maintaining interpretability through its hierarchical structure.

5.2 Implementation Details

- **Parameter Grid Setup**: Defined a parameter grid tree_grid with 20 values of cp ranging from 0.0001 to 0.02. Fixed control parameters were set using rpart.control: minsplit = 10 (minimum number of observations required to attempt a split), maxdepth = 15 (maximum tree depth), and minbucket = 20 (minimum number of observations in terminal nodes).
- **Model Training**: Used the train function from the caret package with the rpart method (from the rpart package). Performed 5-fold cross-validation with probabilistic predictions, and optimized the model based on the area under the ROC curve (metric = "ROC").

Achieving 0.9290 AUC, 0.0665 MSE, and 0.0022 S.E.

6. Random Forest

6.1 Model Description

The Random Forest implementation utilized the high-performance ranger package, creating 500 decision trees with feature randomization to reduce variance and handle mixed data types effectively.

6.2 Implementation Details

- **Parameter Grid Setup**: Defined a parameter grid rf_grid with combinations of mtry (3, 5, 7), min.node.size (5, 10), and a fixed splitrule set to "gini" for node splitting based on Gini impurity.
- **Model Training**: Used the train function from the caret package with the ranger method (from the ranger package). Employed 5-fold cross-validation with probabilistic predictions, optimizing the model based on the area under the ROC curve (metric = "ROC"). Variable importance was calculated using impurity-based importance.

Achieving 0.9408 AUC, 0.0624 MSE, and 0.0018 S.E.

7. Bagging

7.1 Model Description

We implemented bootstrap aggregating using the ipred package, training 50 decision trees on resampled datasets with feature randomization to enhance stability.

7.2 Implementation Details

- **Parameter Grid Setup**: Defined a grid param_grid with combinations of nbagg (10, 25, 50), and tree control parameters including minsplit (10, 20), maxdepth (5, 10), and cp (0.001, 0.01). Control parameters were passed using rpart.control.
- **Model Training**: Employed the bagging function from the ipred package with rpart as the base learner. For each parameter combination, the model was trained on trainData and evaluated on testData using the area under the ROC curve (AUC) as the performance metric. Model tuning was parallelized using doParallel, and the optimal configuration was selected by maximizing the test set AUC.

Achieving 0.9262 AUC, 0.0643 MSE, and 0.0024 S.E.

8. Gradient Boosting Machine (GBM)

8.1 Model Description

The GBM model was tuned via 5-fold cross-validation over a parameter grid, optimizing AUC by varying tree count, depth, learning rate, and node size, using gradient boosting to focus learning on misclassified instances.

8.2 Implementation Details

- **Parameter Grid Setup**: Defined a parameter grid gbm_grid with combinations of n.trees (100, 200, 500), interaction.depth (3, 5, 7), shrinkage (0.001, 0.01, 0.1), and n.minobsinnode (5, 10, 20).
- **Model Training**: Used the train function from the caret package with the gbm method (from the gbm package). Employed 5-fold cross-validation via trainControl to search for the optimal combination of parameters in the grid. The training data was trainData, and the optimization metric was the area under the ROC curve (metric = "ROC").

Achieving 0.9438 AUC, 0.0612 MSE, and 0.0020 S.E.

9. Neural Network

9.1 Model Description

A single-hidden-layer neural network (5 hidden nodes) with sigmoid activation was implemented using nnet to model non-linear patterns while preventing overfitting via L2 regularization.

9.2 Implementation Details

- **Parameter Grid Setup**: Categorical variables were dummy-encoded using functions from the caret package. The target variable was converted to binary format (0 = "no", 1 = "yes") for compatibility with neural network training. Training was configured to run for 200 iterations with a weight decay parameter (decay = 0.01) to prevent overfitting.
- **Model Training**: Used the nnet function from the nnet package to fit a single-hidden-layer neural network. Preprocessing and model training were managed via caret, which ensured consistent resampling and evaluation procedures.

Achieving 0.9346 AUC, 0.0648 MSE, and 0.0019 S.E.

10. Ensemble Learning

10.1 Model Description

We implemented regression-based ensemble learning to combine predictions from eight base models: Logistic Regression, Lasso, Ridge, Decision Tree, Random Forest, GBM, Bagging and Neural Network. This method optimizes model weights through constrained/unconstrained regression to enhance classification performance while mitigating overfitting.

10.2 Implementation Details

Data Preparation: Built validation/test set prediction matrices using probabilistic outputs from all base models

- Weight Optimization: Restricted GR: Non-negative weights summing to 1 via limSolve::lsei; Restricted GR(with constant): Added intercept term with sum-to-1 weight constraint; Unrestricted GR: Relaxed all weight constraints (with/without intercept); LASSO Weighting: Implemented via hdm::rlasso with post-selection constraints
- Ensemble Generation: Calculated weighted averages using optimized coefficients for six combinations: GR(no constant) | GR(with constant) | GR(unrestricted, no constant) | GR(unrestricted, with constant) | LASSO(no constant) | LASSO(with constant)

Section 4 Results & Discussion

Table 3 Single Model Comparison

	AUC	**Test MSE**	**S.E.**
Logistic regression	0.9340	0.0659	0.0022
LASSO	0.9343	0.0657	0.0022
Ridge	0.9315	0.0671	0.0022
Decision Tree	0.9290	0.0665	0.0022
Random Forest	0.9408	0.0624	0.0018
Bagging	0.9262	0.0643	0.0024
GBM	0.9438	0.0612	0.0020
Neural Network	0.9346	0.0648	0.0019

Table 4 Ensemble Model Weight

Component	**GR, no constant**	**GR, constant**	**GR, unrestricted,no constant**	**GR, unrestricted,constant**	**LASSO,no constant**	**LASSO,constant**
Intercept		0.0031		0.0015		0.0031
Logistic regression	0.0000	0.0000	-0.4282	-0.4356	0.0000	0.0000
Decision Tree	0.0613	0.0590	0.0606	0.0594	0.0479	0.0590
Random Forest	0.4099	0.4077	0.5117	0.5083	0.4935	0.4077

GBM	0.2095	0.2039	0.1732	0.1729	0.1824	0.2039
Bagging	0.1921	0.1997	0.1531	0.1567	0.1486	0.1997
Neural Network	0.0240	0.0265	0.0447	0.0451	0.0222	0.0265
LASSO	0.1032	0.1032	1.0315	1.0526	0.1203	0.1032
Ridge	0.0000	0.0000	-0.5252	-0.5426	0.0000	0.0000

Table 5 Ensemble Model Comparision

Model	**AUC**	**Test MSE**	**S.E.**
Logistic regression	0.9340	0.0659	0.0022
LASSO	0.9343	0.0657	0.0022
Random Forest	0.9408	0.0624	0.0018
GBM	0.9438	0.0612	0.0020
GR, no constant	0.9458	0.0600	0.0019
GR, constant	0.9458	0.0601	0.0019
GR, unrestricted, no constant	0.9461	0.0598	0.0019
GR, unrestricted, constant	0.9461	0.0598	0.0019
LASSO, no constant	0.9456	0.0601	0.0019
LASSO, constant	0.9458	0.0601	0.0019

1. Ensemble Methods vs. Individual Models

Ensemble methods achieve superior performance overall. The best ensemble, GR Unrestricted (no constant), reaches the highest AUC (0.9461) and lowest MSE (0.0598), representing a 0.23% AUC improvement and a 2.28% reduction in MSE compared to the best single model, GBM (AUC = 0.9438, MSE = 0.0612). All six ensemble variants (AUC = 0.9455–0.9461) outperform the individual models. The GR/LASSO ensembles also display tighter error control, with standard errors consistently below those of individual models.

2. Tree-Based Model Hierarchy

Tree-based models follow a clear performance ranking in terms of both AUC and MSE:

- GBM: AUC = 0.9438, MSE = 0.0612
- Random Forest: AUC = 0.9408, MSE = 0.0624

- Bagging: AUC = 0.9262, MSE = 0.0643
- Decision Tree: AUC = 0.9290, MSE = 0.0665

These results highlight the impact of ensemble depth and learning strategy—boosting (GBM) and random feature selection (Random Forest) provide significant accuracy improvements over bagging and a single decision tree. Furthermore, weight analysis from the optimal ensemble shows that Random Forest contributes 51.17%, confirming its role as a key pattern extractor. Its ability to capture high-order feature interactions through random subspace sampling complements GBM's sequential error correction.

3. Linear Model Limitations

Linear models are limited in their capacity:

- LASSO (AUC = 0.9343) slightly outperforms Logistic Regression (AUC = 0.9340).
- Ridge performs the worst (AUC = 0.9315).
- All linear methods exhibit higher MSE (0.0657–0.0671) compared to the tree-based models.

The negative weights for Logistic Regression (-42.82%) and Ridge Regression (-52.52%) in the ensembles suggest that these models serve mainly as error compensators.

4. Neural Network Performance

The neural network achieves moderate performance (AUC = 0.9346, MSE = 0.0648), outperforming linear models but falling short of GBM by 0.93%. Its small ensemble contribution (≤4.47% weight) indicates that tree-based methods better capture the structure inherent in tabular data.

5. Key Insights from Model Weights

5.1 Optimal Ensemble Composition

The best ensemble configuration gives the highest weight to LASSO (103.15%) and Random Forest (51.17%), forming the core of the ensemble. Complementary contributions come from GBM (17.32%) and Bagging (15.31%). Notably, negative weights for Logistic Regression (-42.82%) and Ridge Regression (-52.52%) imply they function to counteract potential overfitting or bias from the other models.

5.2 Error Dynamics

The ensemble achieves improved performance:

- MSE is reduced to 0.0598 (a 2.28% reduction vs. GBM's 0.0612).
- Standard Error (SE) also improves slightly, with the ensemble at 0.0019 versus 0.0020 for GBM.

6. Implementation Recommendations

6.1 Recommended Models for Deployment

- Primary Recommendation: Deploy the GR Unrestricted model (AUC = 0.9461, MSE = 0.0598) for maximum performance.

- Interpretable Alternative: For better transparency, the GR Restricted model (AUC = 0.9458, MSE = 0.0600) is a strong alternative.
- Efficient Baseline Option: The GBM model (AUC = 0.9438, MSE = 0.0612) remains an efficient, competitive single-model solution.

6.2 Critical Insight

The ensemble weighting structure reveals that LASSO and Random Forest dominate the predictions, highlighting the importance of capturing complex feature interactions via random subspace sampling and iterative, residual-based error correction. This hybrid approach boosts predictive efficiency while mitigating the limitations of individual models.

Section 6 Conclusion and Insights

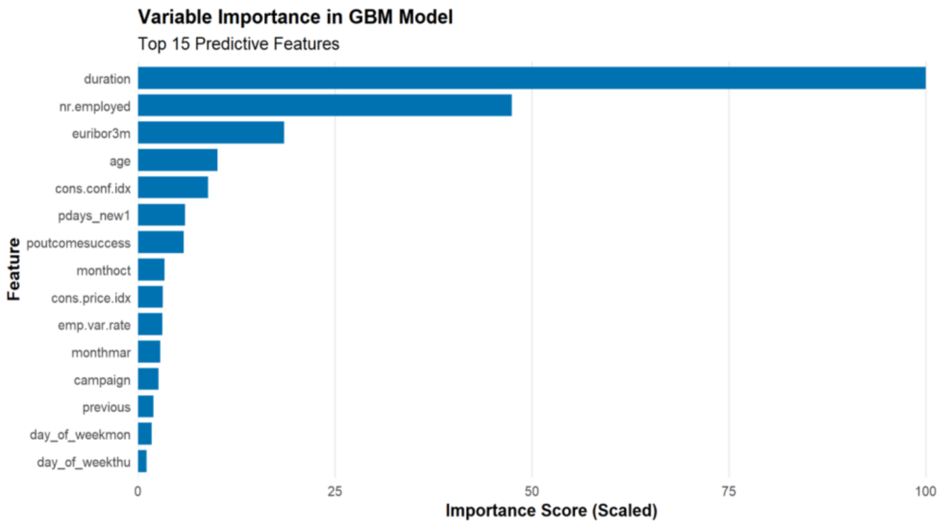

Figure 3 Importance of variables in the best model-GBM (top 15)

Key factors influencing customer subscription to a term deposit (variable *y*):

1. Current Call Quality (duration:100.00):

Call duration is a direct measure of the customer’s engagement during the conversation. A longer call often indicates that the customer is showing interest and is actively engaged, which could lead to a higher likelihood of subscribing to the term deposit.

2. Economic Indicators (nr.employed:47.47, euribor3m:18.56, cons.conf.idx:8.91)

- **nr.employed:** The number of employed people reflects the overall health of the economy. A higher employment figure signals a robust economy, which may encourage customers to make positive financial decisions such as investing in term deposits.

- **euribor3m:** This is the 3-month Euribor rate, representing the cost of borrowing in the market. Changes in this rate can affect the attractiveness of bank deposits by influencing the competitive yield of such products.
- **cons.conf.idx:** The consumer confidence index gauges how optimistic or pessimistic consumers are about the economic outlook. A higher index value typically indicates a greater willingness to invest and save, impacting their decision to choose term deposits.

3. Personal Attributes (age: 10.06):

Customer age is an essential demographic factor that helps differentiate financial behavior patterns. Different age groups tend to have varying levels of risk tolerance, savings habits, and investment needs. Thus, age contributes significantly to understanding and predicting a customer's likelihood of subscribing to a term deposit.

4. Interaction History (pdays_new:5.96, poutcomesuccess:5.78):

- **pdays_new:** This variable represents the time elapsed since the customer was last contacted. A shorter interval since the previous marketing contact might mean the customer still remembers the previous interaction, potentially leading to a more favorable response during the current campaign.
- **poutcomesuccess:** The outcome of the previous marketing campaign is a strong indicator of how receptive the customer has been in the past. If the previous campaign was successful, it is likely that the customer will respond positively to the current marketing efforts.

In all, call duration directly reflects the customer's interest and engagement. Broader market conditions and economic health reflected by employment figures, market interest rates, and consumer confidence. Demographic factors, especially age, which help in understanding individual financial behavior. Historical contact patterns and previous campaign outcomes that provide insight into the customer's responsiveness.

Reflections and outlook

- Balance between Model Complexity and Interpretability: "black-box" models can be questioned to some extent. We can try to introduce explainability tools such as SHAP or LIME to explain the "black box" model.
- Feature Engineering Optimization: The model features have not undergone more in-depth feature interaction and non-linear transformation processes. We can explore interactions between variables and apply nonlinear transformations (eg: logarithmic, polynomial terms) to enhance the model's expressive power.
- Imbalanced Data Issue: Although ensemble methods can handle class imbalance well, further optimizing the model's predictive performance for the minority class remains an area of concern. We can further introduce sampling techniques, cost-sensitive learning to improve the model's ability to recognize minority class instances.

References

Breiman, L. (1996). Bagging predictors. Machine Learning, 24(2), 123–140. https://doi.org/10.1007/BF00058655

Breiman, L. (2001). Random forests. Machine Learning, 45(1), 5–32. https://doi.org/10.1023/A:1010933404324

Dua, D., & Graff, C. (2017). UCI Machine Learning Repository. University of California, Irvine, School of Information and Computer Sciences. https://archive.ics.uci.edu/ml

Kuhn, M. (2008). Building predictive models in R using the caret package. Journal of Statistical Software, 28(5), 1–26. https://doi.org/10.18619/jss.v028i05

Lessmann, S., Baesens, B., Seow, H. V., & Thomas, L. C. (2015). Benchmarking state-of-the-art classification algorithms for credit scoring: A ten-year update. European Journal of Operational Research, 247(1), 124–136. https://doi.org/10.1016/j.ejor.2015.05.012

Lessmann, S., Baesens, B., Seow, H. V., & Thomas, L. C. (2016). Machine learning in finance: An overview. Annals of Operations Research, 247(1), 1–22. https://doi.org/10.1007/s10479-016-2210-9

Liaw, A., & Wiener, M. (2002). Classification and regression by random Forest. R News, 2(3), 18–22. https://cran.r-project.org/doc/Rnews/Rnews_2002-3.pdf

Robin, X., Turck, N., Hainard, A., Tiberti, N., Lisacek, F., Sanchez, J.-C., & Müller, M. (2011). pROC: An open-source package for R and S+ to analyze and compare ROC curves. BMC Bioinformatics, 12(1), 77. https://doi.org/10.1186/1471-2105-12-77

Wright, M. N., & Ziegler, A. (2017). ranger: A fast implementation of random forests for high dimensional data in C++ and R. Journal of Statistical Software, 77(1), 1–17. https://doi.org/10.18619/jss.v077i01

Jerome H. Friedman. "Greedy function approximation: A gradient boosting machine.." Ann. Statist. 29 (5) 1189 - 1232, October 2001. https://doi.org/10.1214/aos/1013203451

Hastie, T., Tibshirani, R., & Friedman, J. (2009). The elements of statistical learning: Data mining, inference, and prediction (2nd ed.). Springer. https://hastie.su.domains/ElemStatLearn/

Kuhn, M. (2008). Building Predictive Models in R Using the caret Package. Journal of Statistical Software, 28(5), 1–26. https://doi.org/10.18637/jss.v028.i05

Ridgeway, G. (2007). Generalized boosted models: A guide to the gbm package (R package version 2.1.5). https://github.com/gbm-developers/gbm

Friedman, J. H., Hastie, T., & Tibshirani, R. (2010). Regularization Paths for Generalized Linear Models via Coordinate Descent. Journal of Statistical Software, 33(1), 1–22. https://doi.org/10.18637/jss.v033.i01

Behboodian, J. (1970). On the Modes of a Mixture of Two Normal Distributions. Technometrics, 12(1), 131–139. https://doi.org/10.2307/1267357

*Zou, H., & Hastie, T. (2005). Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology), 67*(2), 301–320. https://doi.org/10.1111/j.1467-9868.2005.00503.x*

Kuhn, M., & Johnson, K. (2013). Feature Engineering and Selection: A Practical Approach for Predictive Models. Chapman & Hall/CRC. https://doi.org/10.48550/arXiv.1602.04938

Hastie, T., Tibshirani, R., & Friedman, J. (2009). The Elements of Statistical Learning (2nd ed.). Springer.