PREDICTIVE ANALYTICS

"Optimizing Client Engagement and Investment Prediction:

A Machine Learning Approach to Maximizing Expected Profit"

Federico Scandizzo

Paulo Cheng

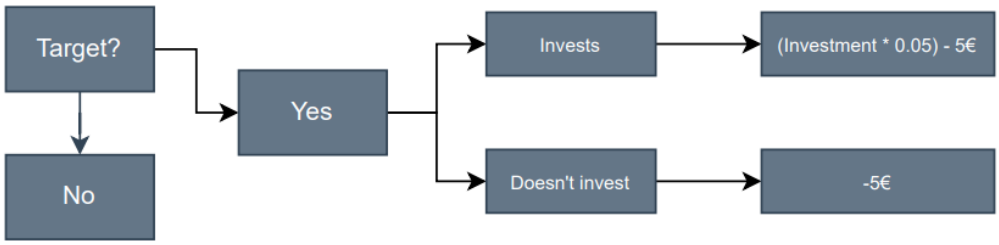
Georg Hoch

1. Introduction

This report is intended for authorized Allianz employees. As a global leader in financial services and insurance, Allianz manages over €1.8 trillion in assets and has a strong presence in the retail client segment. Given the limited outreach capacity of 1,500 contacts per week out of 7,500 available leads, prioritizing high-potential clients through data-driven decision-making is essential. As newly joined analysts in the customer engagement team, we conducted a client outreach optimization project to enhance Allianz's engagement strategies. Using predictive analytics and machine learning, we analyzed historical client interactions to refine our approach. Weekly feedback was incorporated to improve predictions and optimize outreach efficiency. This report presents our final predictive model, developed over the past week for potential firm-wide implementation. It outlines the data sources, modeling process, and predictive techniques used, along with key findings. The objective is to provide decision-makers with an actionable strategy to maximize profitability while reducing acquisition costs. To ensure clarity, the report includes both technical insights and strategic recommendations. We first describe the model development process, followed by the final version and potential paths for future implementation. By leveraging advanced analytics, Allianz can enhance client engagement, allocate resources effectively, and drive long-term business growth.

2. Business Understanding

As previously mentioned, the number of clients to be contacted is at most 1500 out of 7500 every week. Due to this constraint, these 1500 clients will be the highest potential value prospects to be contacted. Allianz's business model is the following: (1) the company earns an average profit of 5.0% on the investments made by clients. (2) Historical data indicates that clients invest ca. 22% of their current balances into financial products. Still, there is a marketing cost associated with each attempt to contact a client of €5. The objective is to not only find the clients that are most likely to invest but also who will invest higher amounts, yielding a higher return for the company.



To choose our target client list, two predictive models will be used:

- A **classification model** first, that will estimate the probability of investment from each client. This model returns a value that we call "*predict_proba_yes*".
- A **regression model** secondly, to predict the expected investment amount of each client ("*predict_investment"*).

Then, the clients are ranked by "*expected_profit*", which is the profit each client is expected to generate to the company. This is computed as a score as follows:

$$expected\_profit = predict\_proba\_yes * (predict\_investment * 0.05) - 5$$

2.1 Data understanding & variables

The dataset contained 11 numerical and categorical variables, including client demographics (ID, age, job, marital), communication preferences (*preferred_contact*), financial indicators (*account balance*), loan details (*loan_house, loan_personal*), marketing history (*n_marketing_contacts*), and interaction metrics (*call_length*). The objective is to build both classification and regression models to predict the investment amount (*investment*). At first, only client investment amount from the first period were known, with 7500 observations. For every consecutive week, more information was gained on the clients previously selected. The received feedback data was biased towards successful contacts. To mitigate this, the model was always tested on a portion of the original 7500 clients from the first period, to see its performance on the general population.

2.2 Exploratory Data Analysis

Given that the datasets obtained from the weeks after are biased towards the model's predictions, a visual inspection will be done only on the dataset from the first week, which is more representative of the population. Key investment determinants such as client *balance* and previous *investments* will be examined based on client demographics. Doing so will provide an essential understanding of patterns and trends.

As Figure 1 shows, clients with a tertiary education are the ones with a notably higher investment mean than the others. Figure 2 shows clients with a management-related job have a significantly higher investment mean than others, while self-employed and admin-related jobs are second best.

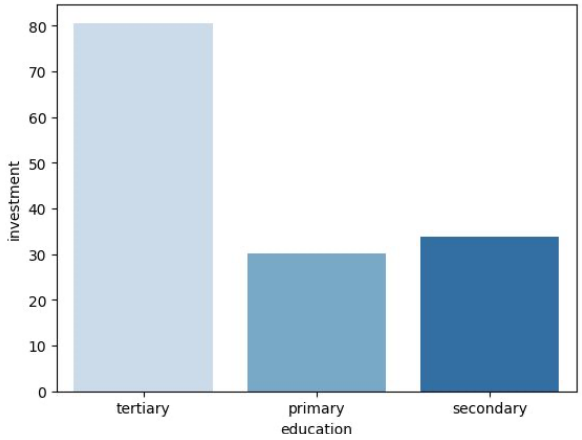


Figure 1

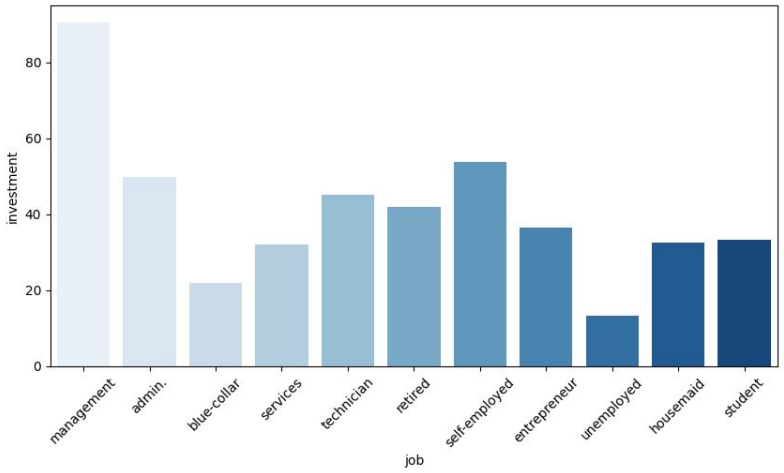


Figure 2

Figure 3 shows clients without a loan on their house have double the investment mean of those who do have it, while Figure 4 shows clients who are single have the highest investment mean.

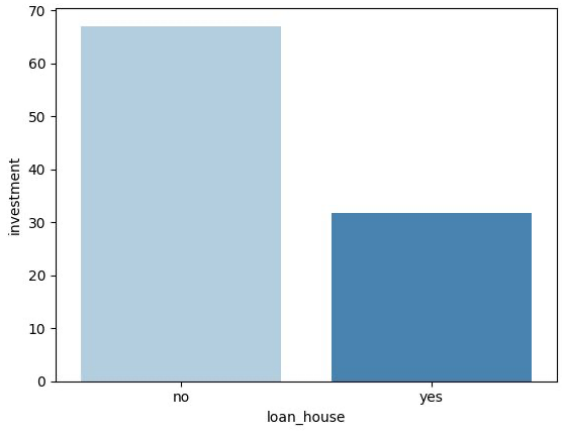


Figure 3

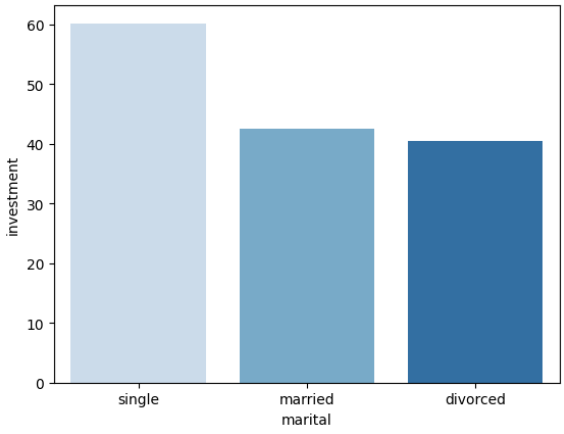


Figure 4

Regarding *balance*, the average is about the same across all marital statuses, with married clients being slightly higher, as shown in Figure 5.

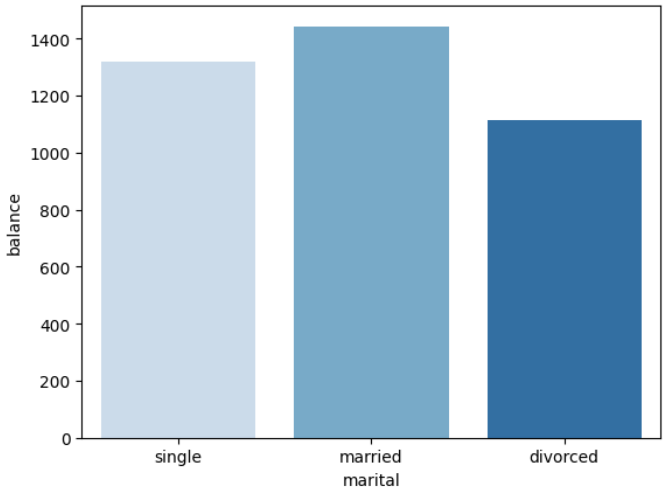


Figure 5

The highest balance mean across jobs is held by retired clients, followed by management-related jobs and entrepreneurs (see Figure 6).

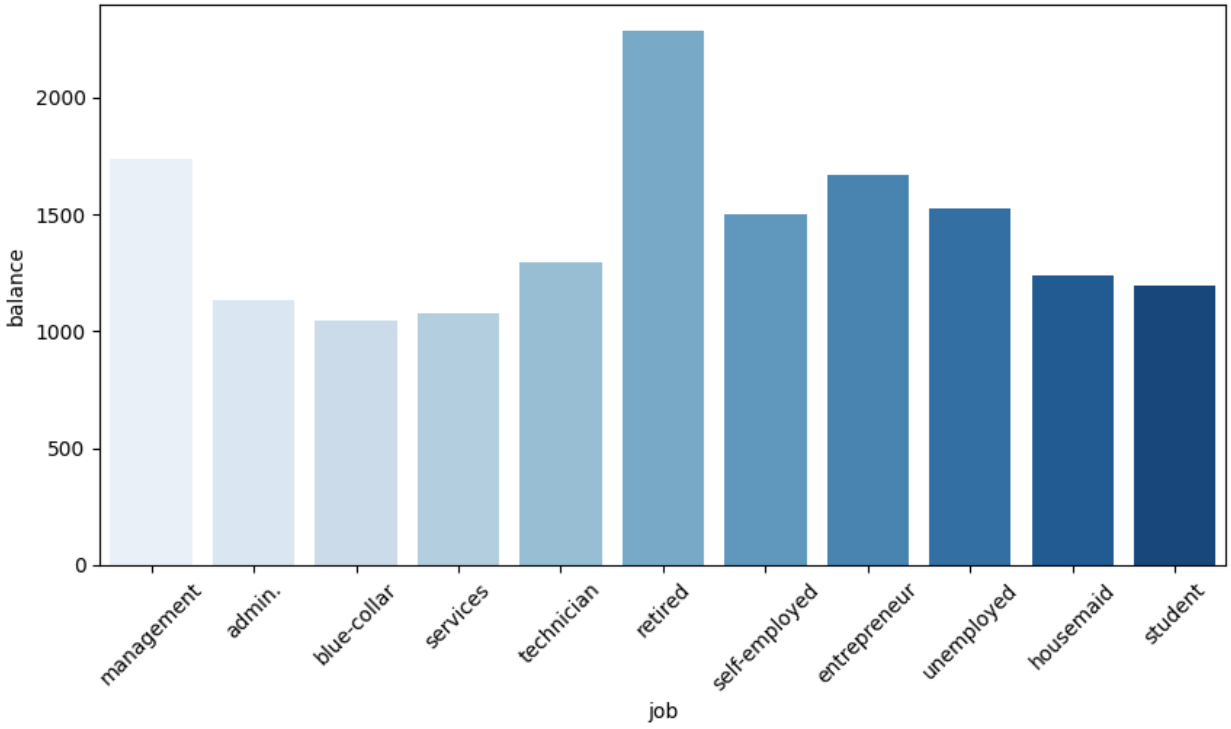


Figure 6

Similar to investment, the highest balance mean is also held by clients with a tertiary education and those without a loan on their house (see Figure 7 & 8).

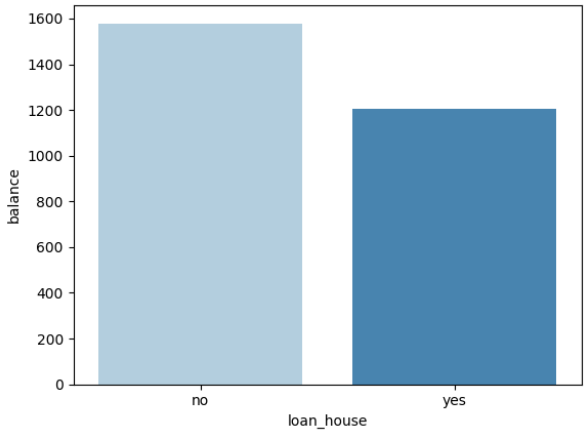


Figure 7

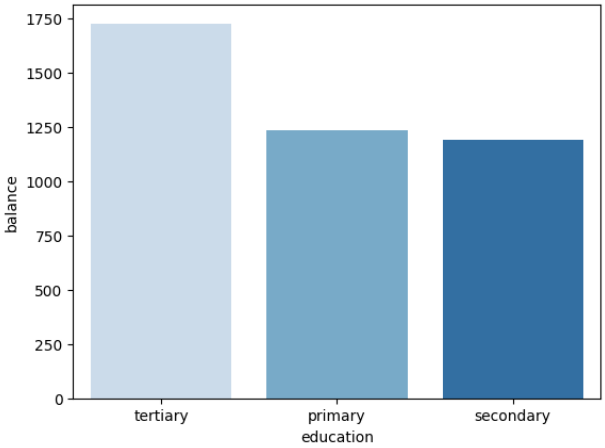


Figure 8

Finally, Figure 9 suggests a strong positive relationship between balance and investment, with the strongest one occurring for clients with a tertiary education. However, the relationship is inconsistent indicating that simple visual analysis is insufficient for drawing accurate conclusions. This highlights the need for machine learning models to identify deeper, non-linear patterns and improve predictive accuracy in the following sections.

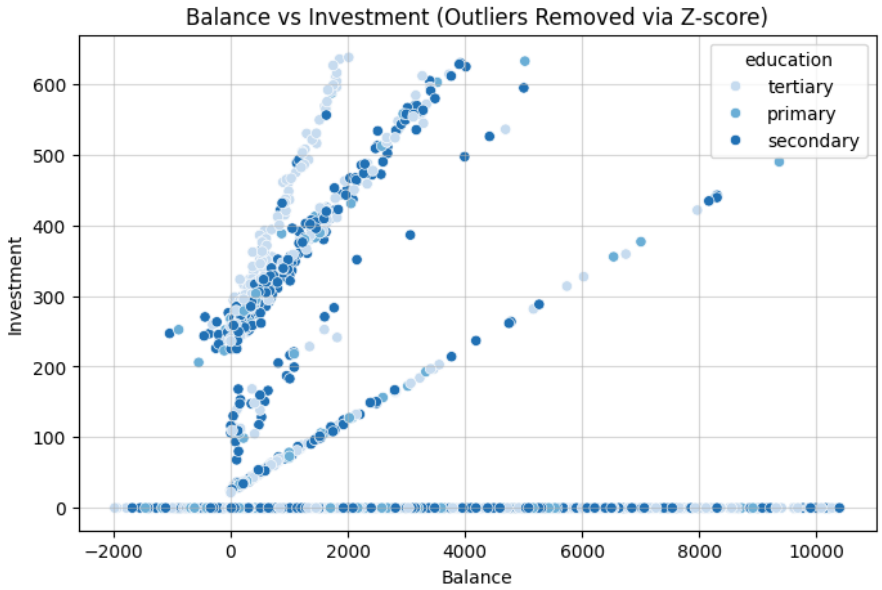
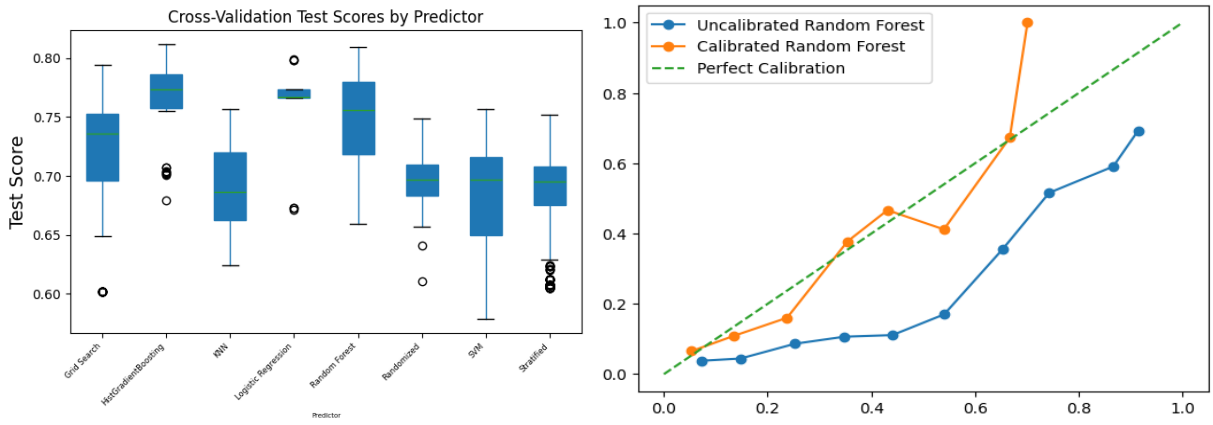


Figure 9

3. Classification model

Given that a classification model returns a categorical outcome variable, some of the variables in our dataset had to be transformed in order to model properly. Columns such as "*call_length*", "*id*", and "*period*", that are not informative or that we don't have information about in the future periods were dropped. We then created a new target variable "*investment_binary*", that gives a binary value to the customer referring to if they had invested in previous periods. Afterwards the data was split into 3 sets: validation (30%), train (40%) and test (30%). To give our model more information, we added the results that we had gotten from the previous weeks into our training set only. For our preprocessing pipeline, we implemented separate transformers for numerical and categorical features. For numeric features we created a numeric transformer pipeline that drops highly correlated columns, imputes missing values with the median, filters out low variance features with a threshold of 0.08, and standardizes the data to ensure consistent scaling across all features. Similarly, for the categorical features, we used a categorical transformer that groups infrequent categories, replaces missing values with a *'missing'* label, encodes categorical variables using one-hot encoding, and applies feature selection to retain only the most relevant features. This way we ensured that the data was properly formatted, scaled, and optimized for model training. After these steps, we tested hyperparameter tuning to a Decision Tree Classifier using three different techniques: *GridSearchCV*, Repeated Stratified K-Fold Cross-Validation combined with *GridSearchCV*, and *RandomizedSearchCV,* all with 5-fold cross-validation, to optimize model performance. The Decision Tree Classifier enables us to inspect feature importance, helping us understand which variables most strongly influence investment behavior. By analyzing tree splits, we can interpret decision-making paths, revealing how different customer attributes impact classification outcomes. We then further experimented with SVC, Random Forest Classifier, KNN Classifier, and Logistic Regression, each fine-tuned using GridSearchCV with 5-fold cross-validation to improve model performance. Lastly, we experimented with Gradient Boosting using the *HistGradientBoostingClassifier*, incorporating hyperparameter tuning also via *GridSearchCV* with 5-fold cross-validation. For every model we trained, we analyzed the ROC curve to evaluate its performance in distinguishing between classes, while also considering the AUC metric. Furthermore, for every model, we computed the confusion matrix to gain a deeper understanding of its classification performance, to assess model accuracy. Lastly, we summarized and visualized the cross-validation results using a boxplot, that provides insight into the distribution of test scores

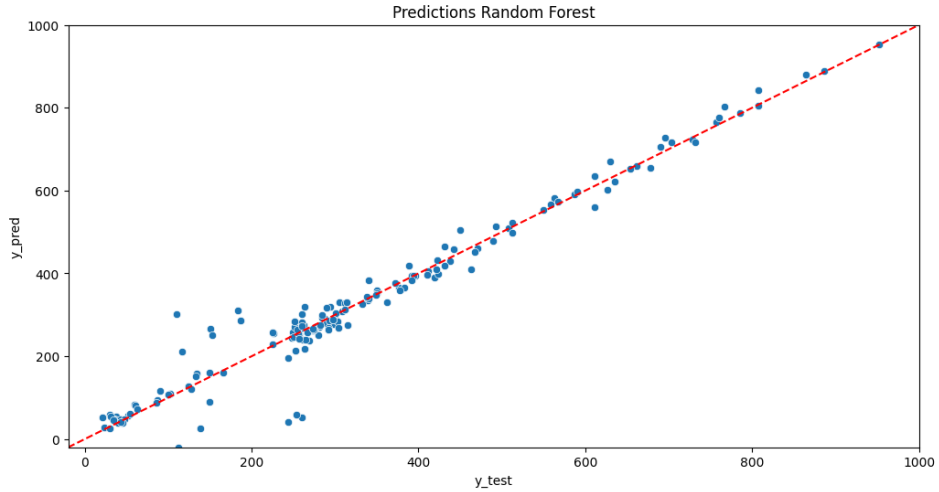
across all models. In the end, we decided to use Random Forest as it had second best score, while proving the most reliable confusion matrix. However, we had to calibrate the model since a Random Forest does not inherently produce well-calibrated probability estimates like a Logistic Regression.



4. Regression Model

For the regression model we had to make a slight change in the data preprocessing part. The same columns were dropped but since we want to model the amount invested no binary category was created and we kept only observations that had investment greater than 0. We then split our data into training and testing only and to our training data we added the results that we got from previous weeks, as usual. For numerical variables, we applied *StandardScaler* to ensure they were standardized, and for categorical variables, we used *OneHotEncoder* to encode them. We tried two regression models, first a Linear Regression and then a Random Forest regression. For the Linear Regression model we introduced polynomial interaction terms using *PolynomialFeatures*, allowing us to capture potential nonlinear relationships in the data and then we assessed its performance using key regression metrics. The Mean Absolute Error (*MAE*) was 23.96, Mean Squared Error (*MSE*) was 1851.88, the Root Mean Squared Error (*RMSE*) was 43.03 and the Median Absolute Error (*MedAE*) was 11.94. Finally, the R^2 score was 0.985, meaning that 98.5% of the variance in the dependent variable was explained by the model, suggesting a strong fit. These results indicate that Linear Regression performed well, capturing most of the variance in the data. However, given potential nonlinear relationships, we also explored Random Forest Regression, which could provide improved predictive accuracy by capturing more complex

patterns. The Random Forest model outperformed the Linear Regression model in terms of error metrics. *MAE* was reduced to 13.47, *MSE* decreased significantly to 439.66, leading to a lower *RMSE* of 20.97, which indicates that large errors had less impact. The *MedAE* was 9.60, and the R^2 score improved further to 99.6%. As a result, we chose the Random Forest model as well for our regression task.



5. Final Strategy

As mentioned previously, our final deployment strategy uses the calibrated classification model to predict the probability of investment and the regression model to predict the expected investment amount and we rank the clients based on a score, the "*expected_profit*". This approach was tested out-of-sample in a dataset where we had to make a prediction and decide which clients to call. We then did the cumulative response curve (left plot) that shows total profit as a function of number of called clients and the estimated profit curve (right plot) for this dataset and concluded that the best number of clients to target would be 850, with an expected profit of 5365€. This strategy maximizes profits by selecting the most profitable clients while optimizing resource allocation.