Predicting customer response to a bank term deposit campaign using classification models

Table of Content

1 Introduction

The objective of this report is to derive insights from a bank campaign promoting a fixed-term savings account by developing a response model to analyse the key features influencing customer responses, using Python. The report focuses on building and evaluating response model using three classification methods: Logistic Regression, Decision Tree, and Random Forest. The aim is to identify the best-performing classification method for maximizing the response model's effectiveness and predictive performance

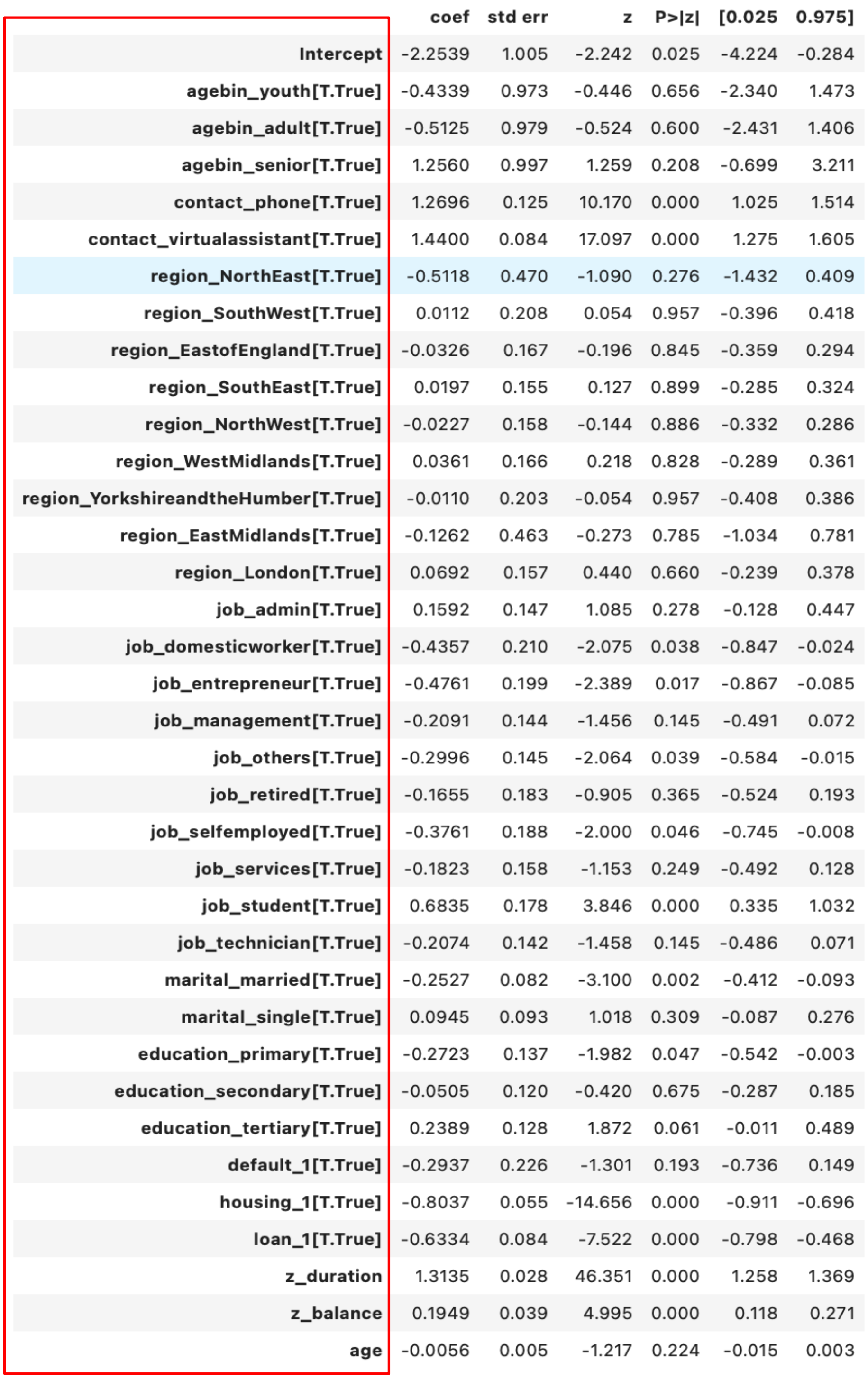
To develop response model, historical data containing campaign contact information and customer response outcomes will undergo pre-processing and exploratory data analysis (EDA). The three classification methods—Logistic Regression, Decision Tree, and Random Forest—will be applied and their performance compared using metrics such as the confusion matrix, ROC curve-AUC, and cumulative gain chart.

2 Response Model

2.1 Logistic Regression

2.1.1 Feature Selection and Parameter

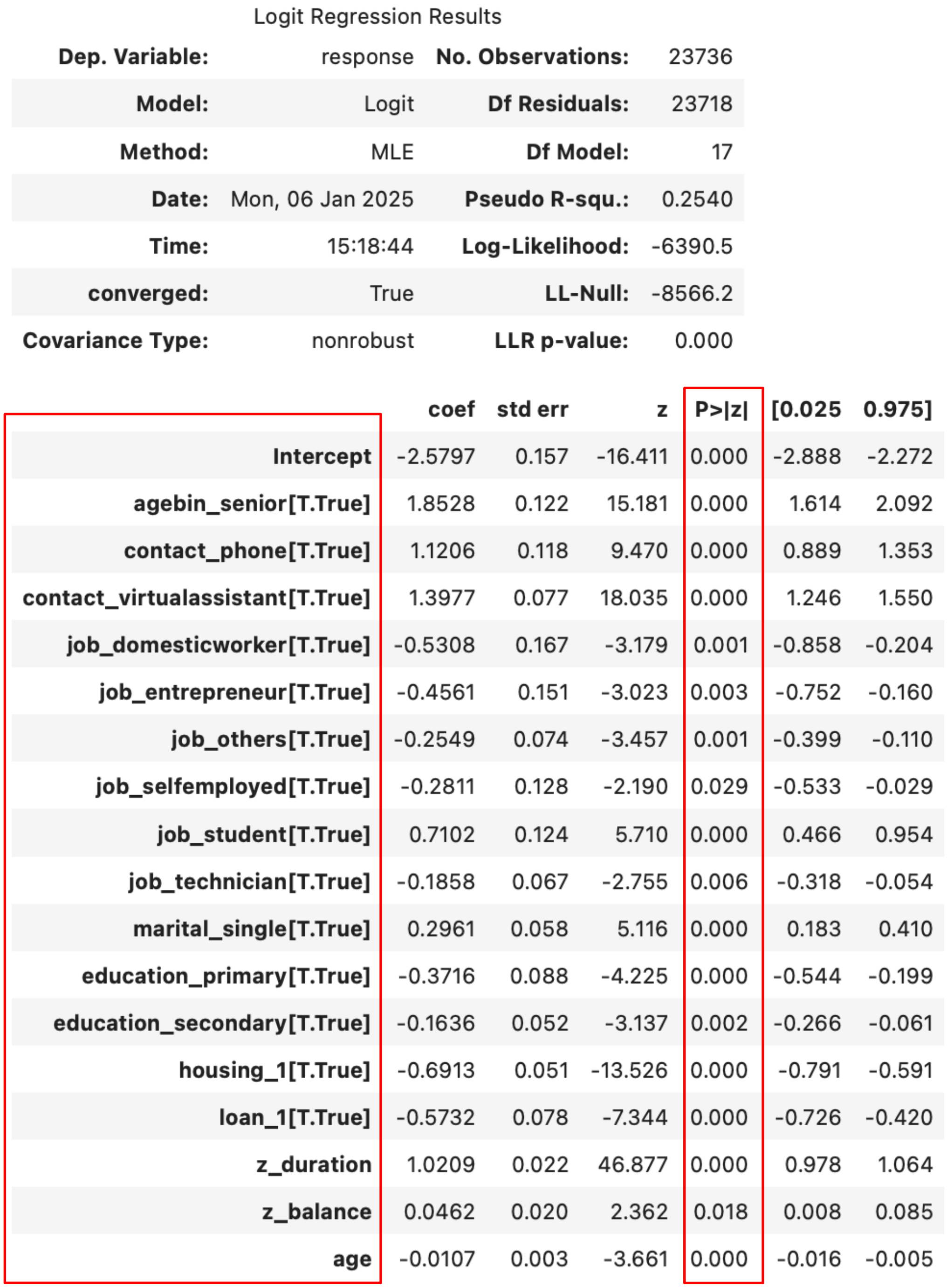
In the feature selection process, the initial logistic regression model included categorical variables such as '*contact'*, *'region'*, *'job'*, and others, while excluding features with missing or irrelevant values. Variables such as *'response'*, *'custID'*, *'duration'*, and '*balance'* were removed due to their distinct purposes or redundancy. These decisions helped in refining the model by focusing on the most relevant predictors, as illustrated in *Figure 1*. This iterative process of selecting features ensures that only meaningful and statistically significant variables contribute to the model, enhancing its predictive performance.



	coef	std err	z	P>|z|	[0.025	0.975]
Intercept	-2.2539	1.005	-2.242	0.025	-4.224	-0.284
agebin_youth[T.True]	-0.4339	0.973	-0.446	0.656	-2.340	1.473
agebin_adult[T.True]	-0.5125	0.979	-0.524	0.600	-2.431	1.406
agebin_senior[T.True]	1.2560	0.997	1.259	0.208	-0.699	3.211
contact_phone[T.True]	1.2696	0.125	10.170	0.000	1.025	1.514
contact_virtualassistant[T.True]	1.4400	0.084	17.097	0.000	1.275	1.605
region_NorthEast[T.True]	-0.5118	0.470	-1.090	0.276	-1.432	0.409
region_SouthWest[T.True]	0.0112	0.208	0.054	0.957	-0.396	0.418
region_EastofEngland[T.True]	-0.0326	0.167	-0.196	0.845	-0.359	0.294
region_SouthEast[T.True]	0.0197	0.155	0.127	0.899	-0.285	0.324
region_NorthWest[T.True]	-0.0227	0.158	-0.144	0.886	-0.332	0.286
region_WestMidlands[T.True]	0.0361	0.166	0.218	0.828	-0.289	0.361
region_YorkshireandtheHumber[T.True]	-0.0110	0.203	-0.054	0.957	-0.408	0.386
region_EastMidlands[T.True]	-0.1262	0.463	-0.273	0.785	-1.034	0.781
region_London[T.True]	0.0692	0.157	0.440	0.660	-0.239	0.378
job_admin[T.True]	0.1592	0.147	1.085	0.278	-0.128	0.447
job_domesticworker[T.True]	-0.4357	0.210	-2.075	0.038	-0.847	-0.024
job_entrepreneur[T.True]	-0.4761	0.199	-2.389	0.017	-0.867	-0.085
job_management[T.True]	-0.2091	0.144	-1.456	0.145	-0.491	0.072
job_others[T.True]	-0.2996	0.145	-2.064	0.039	-0.584	-0.015
job_retired[T.True]	-0.1655	0.183	-0.905	0.365	-0.524	0.193
job_selfemployed[T.True]	-0.3761	0.188	-2.000	0.046	-0.745	-0.008
job_services[T.True]	-0.1823	0.158	-1.153	0.249	-0.492	0.128
job_student[T.True]	0.6835	0.178	3.846	0.000	0.335	1.032
job_technician[T.True]	-0.2074	0.142	-1.458	0.145	-0.486	0.071
marital_married[T.True]	-0.2527	0.082	-3.100	0.002	-0.412	-0.093
marital_single[T.True]	0.0945	0.093	1.018	0.309	-0.087	0.276
education_primary[T.True]	-0.2723	0.137	-1.982	0.047	-0.542	-0.003
education_secondary[T.True]	-0.0505	0.120	-0.420	0.675	-0.287	0.185
education_tertiary[T.True]	0.2389	0.128	1.872	0.061	-0.011	0.489
default_1[T.True]	-0.2937	0.226	-1.301	0.193	-0.736	0.149
housing_1[T.True]	-0.8037	0.055	-14.656	0.000	-0.911	-0.696
loan_1[T.True]	-0.6334	0.084	-7.522	0.000	-0.798	-0.468
z_duration	1.3135	0.028	46.351	0.000	1.258	1.369
z_balance	0.1949	0.039	4.995	0.000	0.118	0.271
age	-0.0056	0.005	-1.217	0.224	-0.015	0.003

Figure 1 LG_Initial model

An iterative refinement process was carried out to remove statistically insignificant variables with p-values>0.05. By the **third iteration**, the model converged, retaining only significant predictors, as shown in *Figure 2*. The final model, which focuses on these key variables, offers actionable insights for the response model.



Logit Regression Results

Dep. Variable:	response	**No. Observations:**	23736
Model:	Logit	**Df Residuals:**	23718
Method:	MLE	**Df Model:**	17
Date:	Mon, 06 Jan 2025	**Pseudo R-squ.:**	0.2540
Time:	15:18:44	**Log-Likelihood:**	-6390.5
converged:	True	**LL-Null:**	-8566.2
Covariance Type:	nonrobust	**LLR p-value:**	0.000

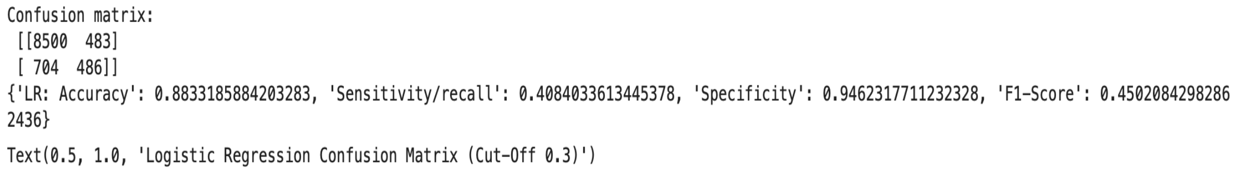
	coef	**std err**	**z**	**P>|z|**	**[0.025**	**0.975]**
Intercept	-2.5797	0.157	-16.411	0.000	-2.888	-2.272
agebin_senior[T.True]	1.8528	0.122	15.181	0.000	1.614	2.092
contact_phone[T.True]	1.1206	0.118	9.470	0.000	0.889	1.353
contact_virtualassistant[T.True]	1.3977	0.077	18.035	0.000	1.246	1.550
job_domesticworker[T.True]	-0.5308	0.167	-3.179	0.001	-0.858	-0.204
job_entrepreneur[T.True]	-0.4561	0.151	-3.023	0.003	-0.752	-0.160
job_others[T.True]	-0.2549	0.074	-3.457	0.001	-0.399	-0.110
job_selfemployed[T.True]	-0.2811	0.128	-2.190	0.029	-0.533	-0.029
job_student[T.True]	0.7102	0.124	5.710	0.000	0.466	0.954
job_technician[T.True]	-0.1858	0.067	-2.755	0.006	-0.318	-0.054
marital_single[T.True]	0.2961	0.058	5.116	0.000	0.183	0.410
education_primary[T.True]	-0.3716	0.088	-4.225	0.000	-0.544	-0.199
education_secondary[T.True]	-0.1636	0.052	-3.137	0.002	-0.266	-0.061
housing_1[T.True]	-0.6913	0.051	-13.526	0.000	-0.791	-0.591
loan_1[T.True]	-0.5732	0.078	-7.344	0.000	-0.726	-0.420
z_duration	1.0209	0.022	46.877	0.000	0.978	1.064
z_balance	0.0462	0.020	2.362	0.018	0.008	0.085
age	-0.0107	0.003	-3.661	0.000	-0.016	-0.005

Figure 2 LG_Final model

2.1.2 Performance Measure

2.1.1.1 Confusion Matrix

The logistic regression model with a cut-off of 0.3 achieved an **accuracy of 88.33%,** indicating strong performance in making correct predictions. However, its **sensitivity of 40.84%** reveals a significant weakness in identifying responders, as more than half of them are missed. The model's high **specificity of 94.62%** indicates it effectively predicts non-responders, avoiding false positives, but this comes at the cost of detecting responders. The **F1-score of 45.02%** shows a moderate balance between precision and recall, but its low sensitivity impacts the score. In summary, the model excels in predicting non-responders but struggles with accurately capturing responders.



```
Confusion matrix:
 [[8500  483]
 [ 704  486]]
{'LR: Accuracy': 0.8833185884203283, 'Sensitivity/recall': 0.4084033613445378, 'Specificity': 0.9462317711232328, 'F1-Score': 0.4502084298286
2436}
Text(0.5, 1.0, 'Logistic Regression Confusion Matrix (Cut-Off 0.3)')
```

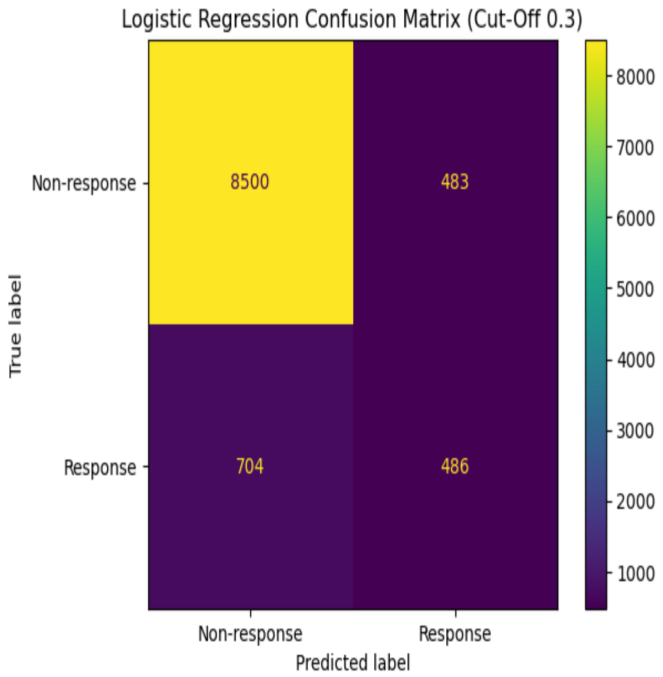


Figure 3 LG_Confusion matrix

2.1.1.2 ROC Curve

The ROC curve of the model, shown in Figure 4, has an **AUC of 0.87**, placing it in the "excellent" category. This performance outperforms random guessing (AUC = 0.5) and indicates an 87% probability of correctly ranking positive instances higher than negative ones. The curve's proximity to the top-left corner highlights the model’s high sensitivity and low false positive rate, demonstrating its effectiveness in identifying positive cases while minimizing false alarms.

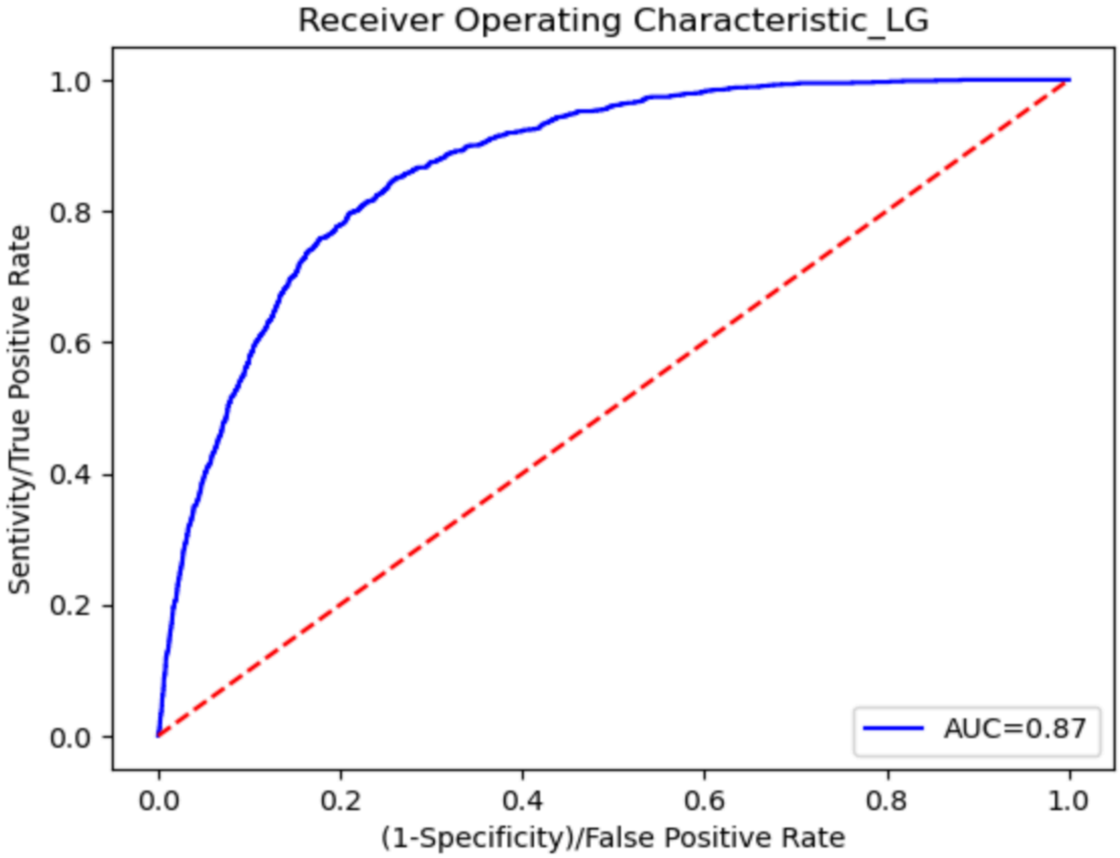


Figure 4 LG_ROC curve

2.1.1.3 *Cumulative Gain Curve*

The cumulative gain curve, shown in Figure 5, assesses the performance of the logistic regression classification model in comparison to random selection. It demonstrates the proportion of responders identified when targeting a percentage of the sample. The blue line represents the baseline, corresponding to random selection, while the orange line illustrates the response model. The curve shows that by targeting just the top **20%** of observations, the model captures over **60%** of responders, and by targeting **70%** of the cases, it captures all responders. This analysis emphasizes that the logistic regression classification model significantly outperforms random selection in effectively identifying responders.

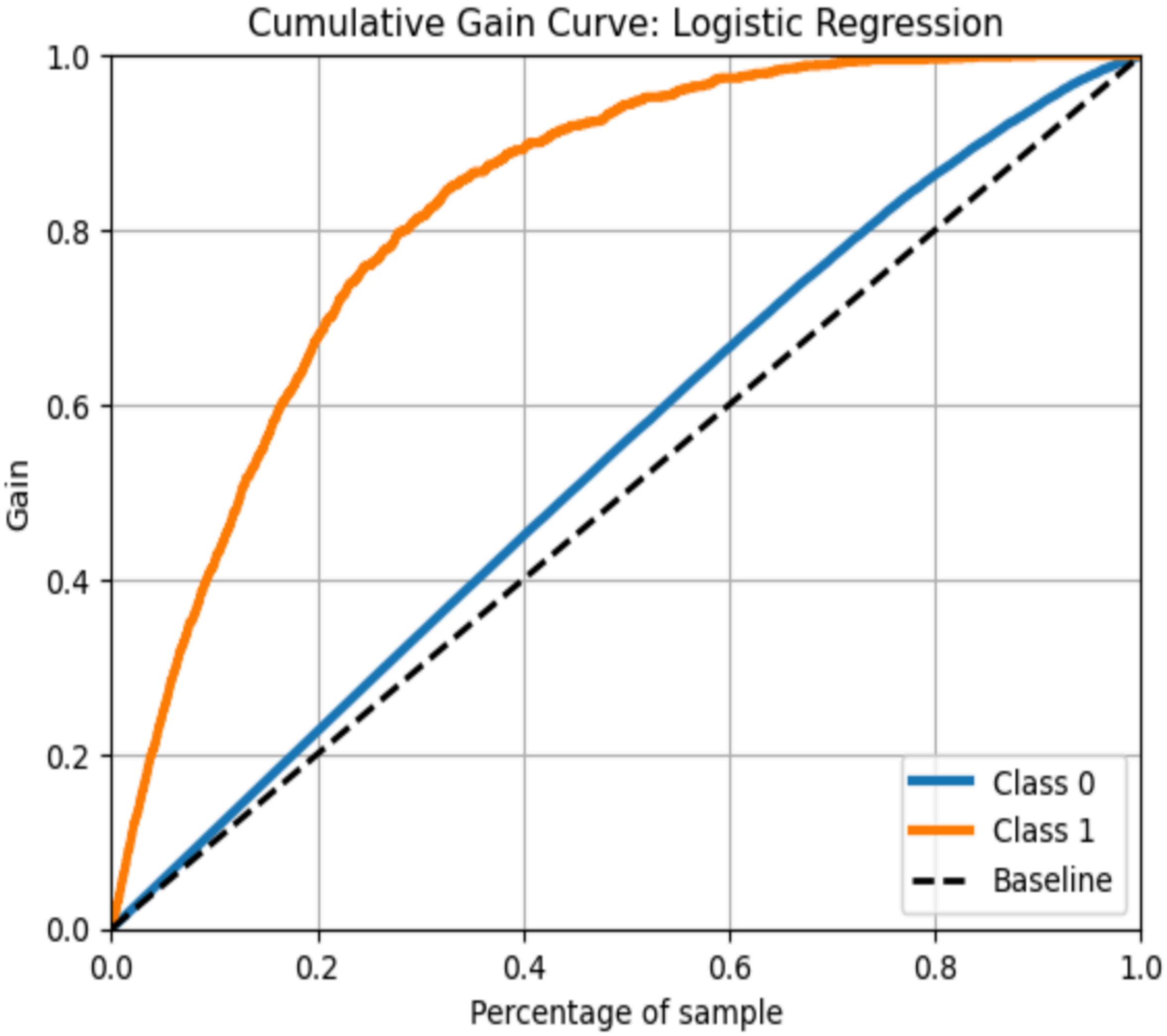


Figure 5 LG_Cumulative gain curve

2.2 Decision Tree

2.2.1 Feature Selection and Parameter

The parameters of the Decision Tree were optimized as follows: the *'max_depth'* was set to **8** to prevent overfitting, and the *'min_samples_split'* was set to **100** to minimize overfitting caused by noise. A fixed ***'****random_state'* of **2222** ensured reproducibility, while the *'criterion'* was set to *'gini'* to enhance classification by minimizing Gini impurity. The dataset was split into **70%** training and **30%** testing subsets, with consistent splits maintained through the random state. The performance of the optimized Decision Tree model is illustrated in *Figure 6.*

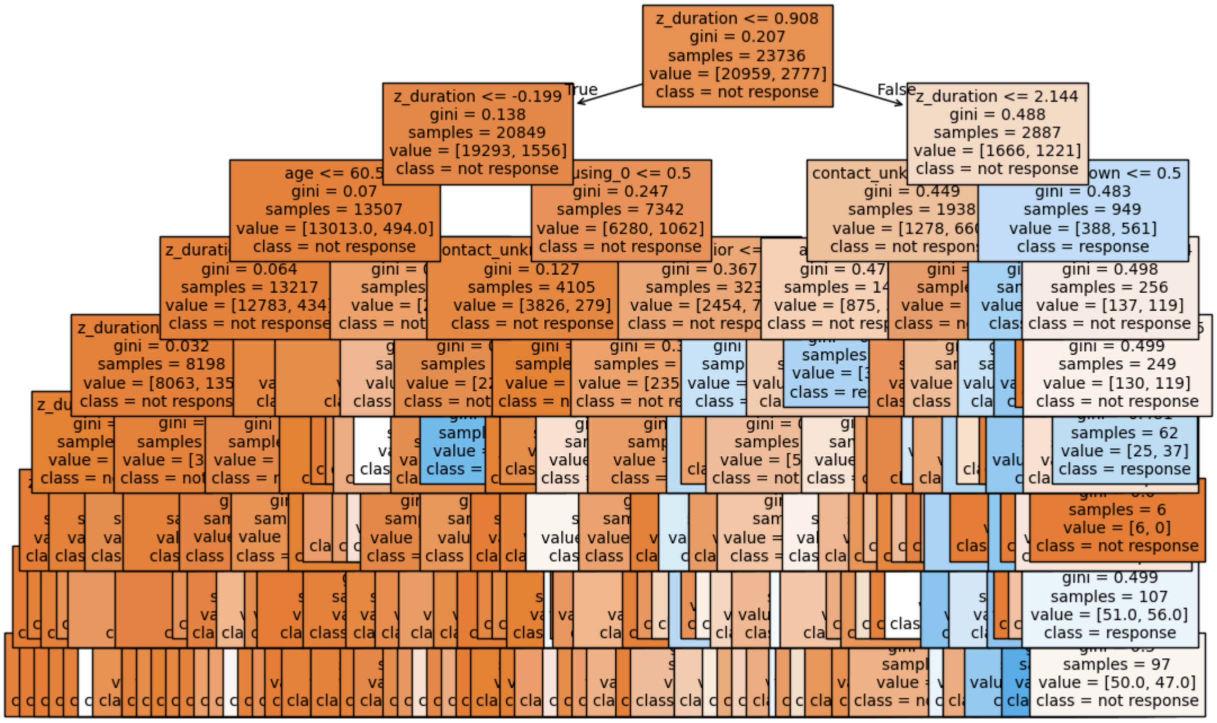
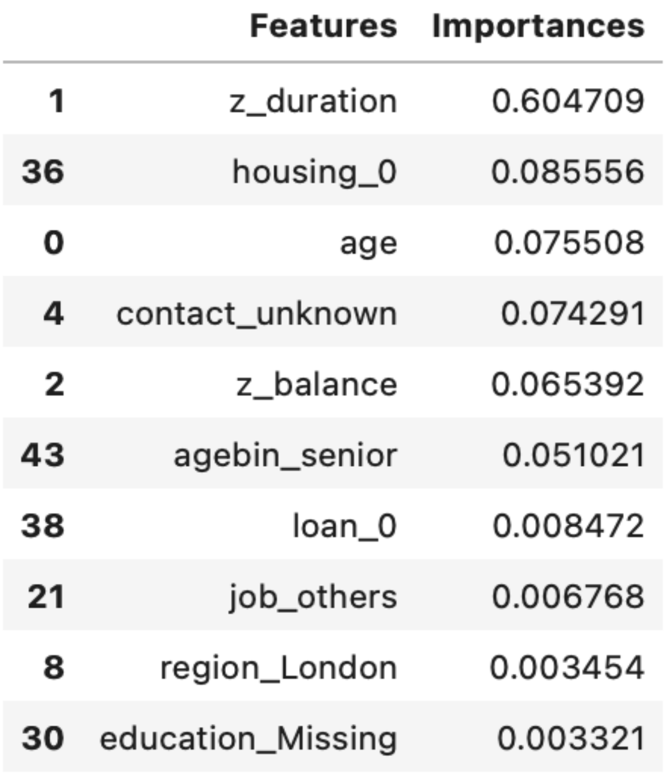


Figure 6 Decision tree model

Figure 7 shows the top 10 features with the highest importance scores, emphasizing the features that contributed most significantly to the model's predictions.



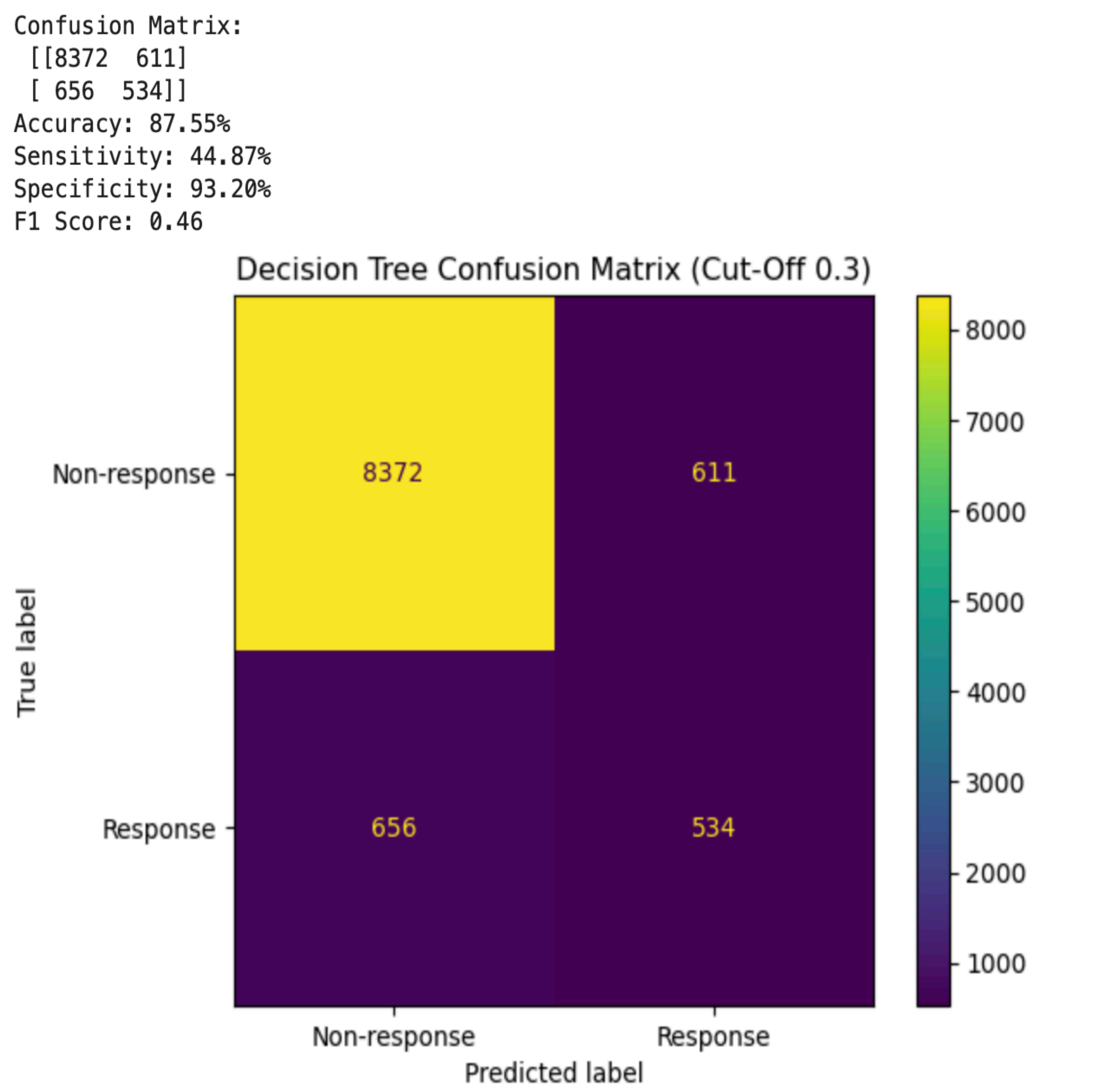
	Features	Importances
1	z_duration	0.604709
36	housing_0	0.085556
0	age	0.075508
4	contact_unknown	0.074291
2	z_balance	0.065392
43	agebin_senior	0.051021
38	loan_0	0.008472
21	job_others	0.006768
8	region_London	0.003454
30	education_Missing	0.003321

Figure 7 DT_features with highest importance scores

2.2.2 Performance Measure

2.2.2.1 Confusion Matrix

The Decision Tree model achieved an **accuracy of 87.55%**, demonstrating its ability to correctly predict the majority of cases. However, its **sensitivity of 44.87%** highlights a challenge in identifying responders, as it misses over half of them. While the model shows strong **specificity (93.20%)** and effectively predicts non-responders with few false positives, it struggles with responders. The **F1 Score of 0.46** indicates a moderate balance between precision and recall but reflects limitations in capturing responders. Overall, the model excels at predicting non-responders but requires optimization to improve responsiveness prediction.



```
Confusion Matrix:
 [[8372  611]
 [ 656  534]]
Accuracy: 87.55%
Sensitivity: 44.87%
Specificity: 93.20%
F1 Score: 0.46
```

Figure 8 DT_Confusion matrix

2.2.2.2 ROC Curve

The ROC curve of the Decision Tree model, shown in *Figure 9*, has an **AUC of 0.86**, which categorizes it as "excellent." This performance surpasses random guessing (AUC = 0.5) and suggests that there is an 86% likelihood of correctly ranking positive instances higher than negative ones. The curve's closeness to the top-left corner emphasizes the model's strong sensitivity and low false positive rate, showcasing its ability to accurately identify positive cases while minimizing false alarms.

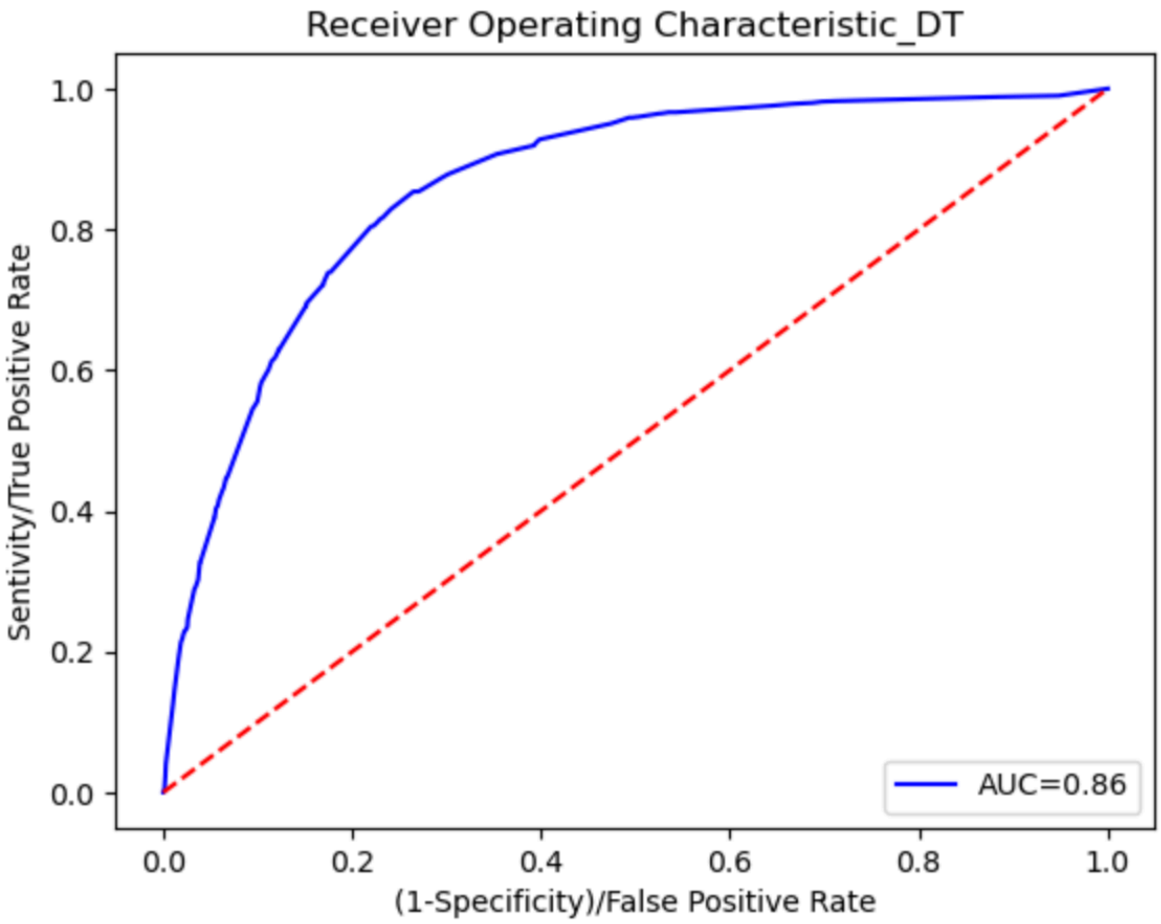


Figure 9 DT_ROC curve

2.2.2.3 Cumulative Gain Curve

The cumulative gain curve for the Decision Tree model, shown in *Figure 10*, shows a sharp increase early on, illustrating the model's efficiency in identifying a significant portion of positive responses with minimal effort. The blue line, representing random selection, serves as a baseline for comparison. The orange line, representing the response model, reveals that by targeting just the top **20%** of observations, over **60%** of responders are captured. By targeting **80%** of the cases, the model successfully identifies all responders. This analysis highlights that the Decision Tree model significantly outperforms random selection in efficiently identifying responders.

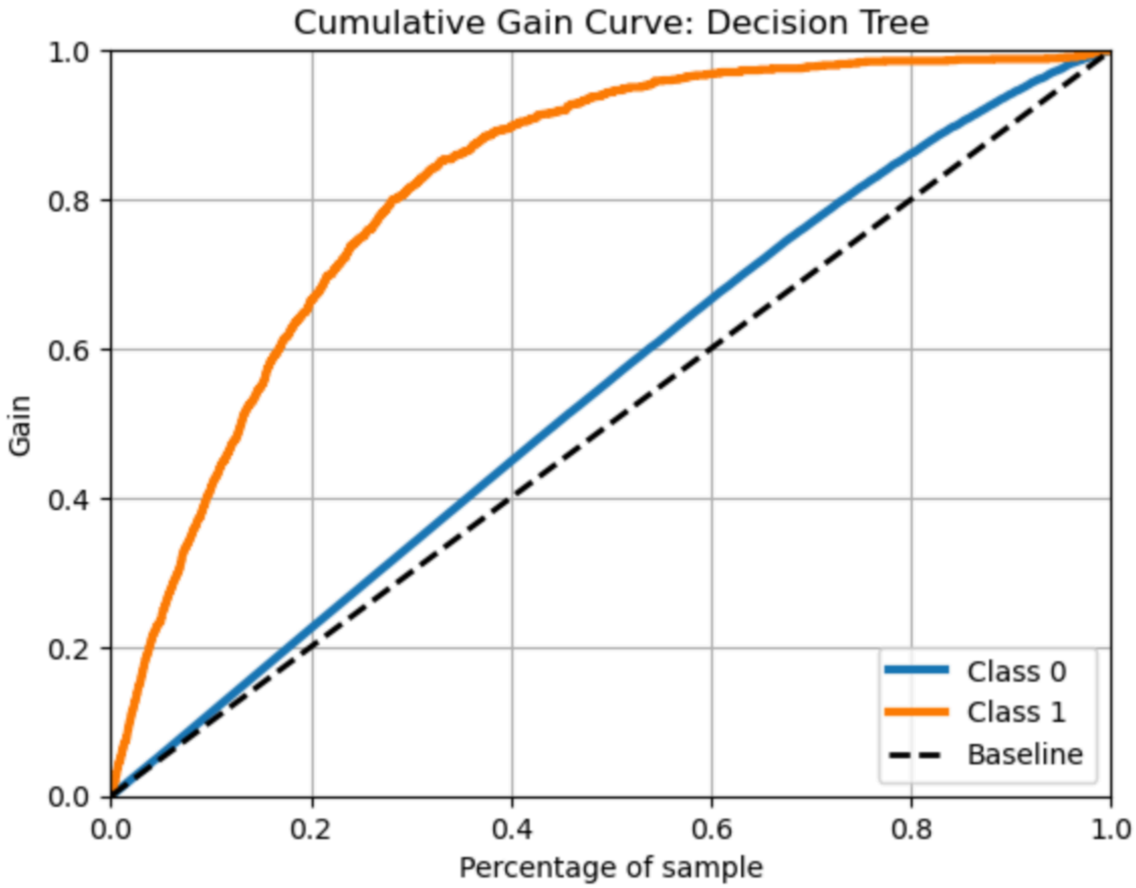
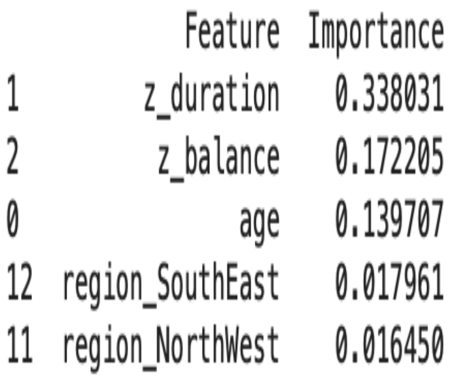


Figure 10 DT_Cummulative gain curve

2.3 Random Forest

2.3.1 Feature Selection and Parameter

The Random Forest model was configured with a *'random_state'* of **2222** for reproducibility and a classification cut-off value of **0.3**, which defines the threshold for categorizing observations as responders or non-responders. The key features influencing the model's predictions are highlighted in *Figure 11*, emphasizing the variables that significantly enhance its performance and accuracy.



```
              Feature  Importance
1          z_duration    0.338031
2           z_balance    0.172205
0                 age    0.139707
12  region_SouthEast    0.017961
11  region_NorthWest    0.016450
```

Figure 11 RF_ features with the highest importance scores

Additionally, *Figure 12* presents the selected features ranked in descending order of importance, highlighting those with the greatest impact on the model's predictions. The top-ranked features are the most critical in determining the likelihood of customer responses, offering valuable insights for crafting targeted and effective marketing strategies.

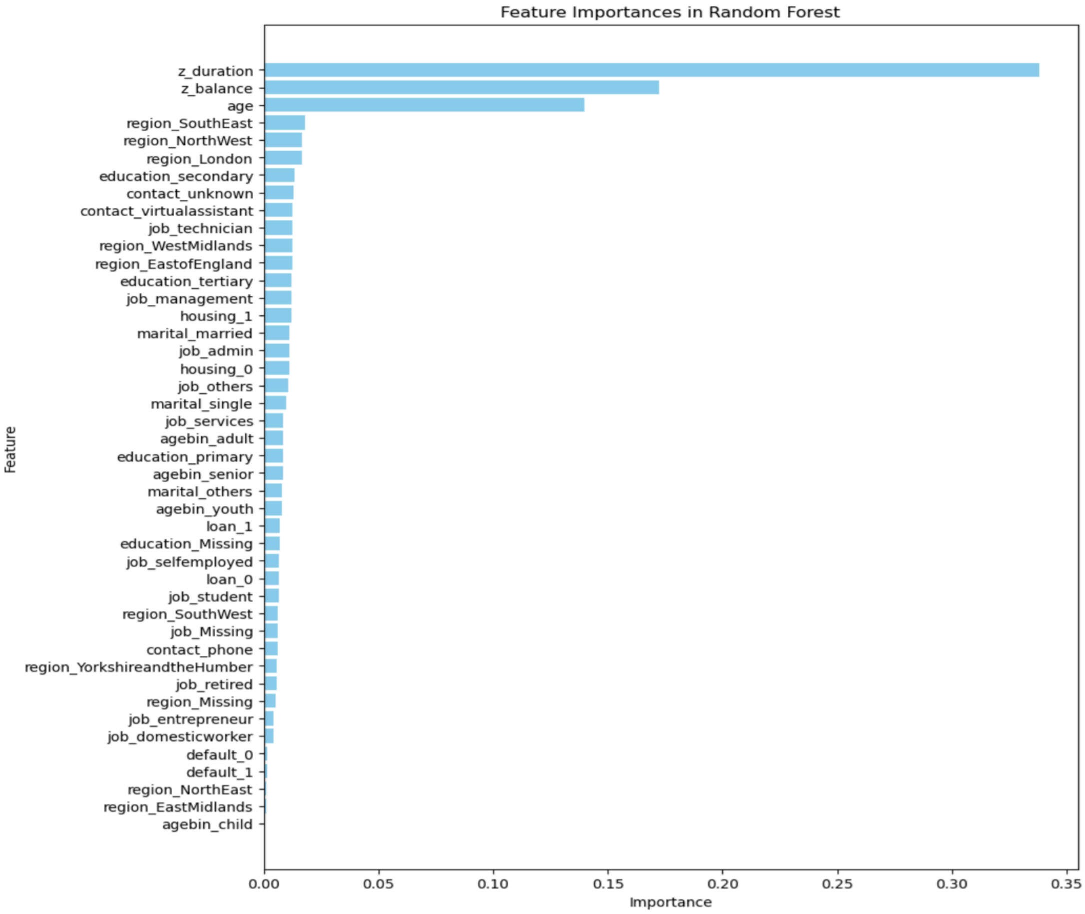
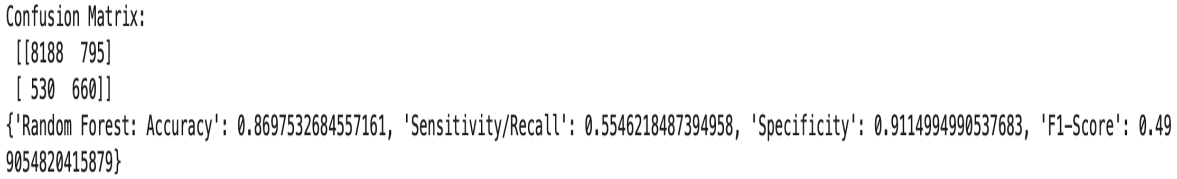


Figure 12 RF_ Important features graph

2.3.2 Performance Measure

2.3.2.1 Confusion Matrix

The Random Forest model demonstrated an **accuracy of 86.98%,** indicating a strong ability to make correct predictions. Its **sensitivity of 55.46%** shows an improvement over previous models, capturing more than half of the actual responders, though there is still room for better detection of positive responses. The model's high **specificity (91.15%)** indicates its effectiveness in predicting non-responders and minimizing false positives. The **F1-score of 0.50** reflects a moderate balance between precision and recall, though further improvements are needed. Overall, the Random Forest model performs better than the logistic regression and decision tree models in identifying responders, but improvements in sensitivity and precision-recall balance are still possible.



```
Confusion Matrix:
 [[8188  795]
 [ 530  660]]
{'Random Forest: Accuracy': 0.8697532684557161, 'Sensitivity/Recall': 0.5546218487394958, 'Specificity': 0.9114994990537683, 'F1-Score': 0.49
9054820415879}
```

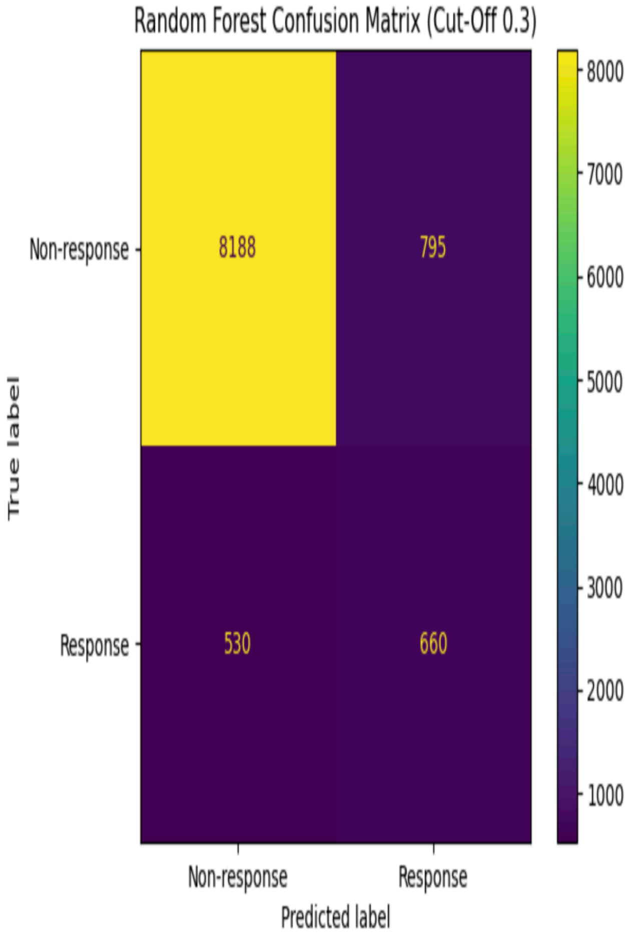


Figure 13 RF_Confusion matrix

2.3.2.2 ROC Curve

The ROC curve of the Random Forest model, shown in *Figure 14*, has an **AUC of 0.87**, indicating an 87% probability of correctly ranking responders above non-responders. This demonstrates the model's strong performance. The curve's proximity to the top-left corner emphasizes the model's high sensitivity and low false positive rate, illustrating its effectiveness in identifying positive cases while minimizing false alarms.

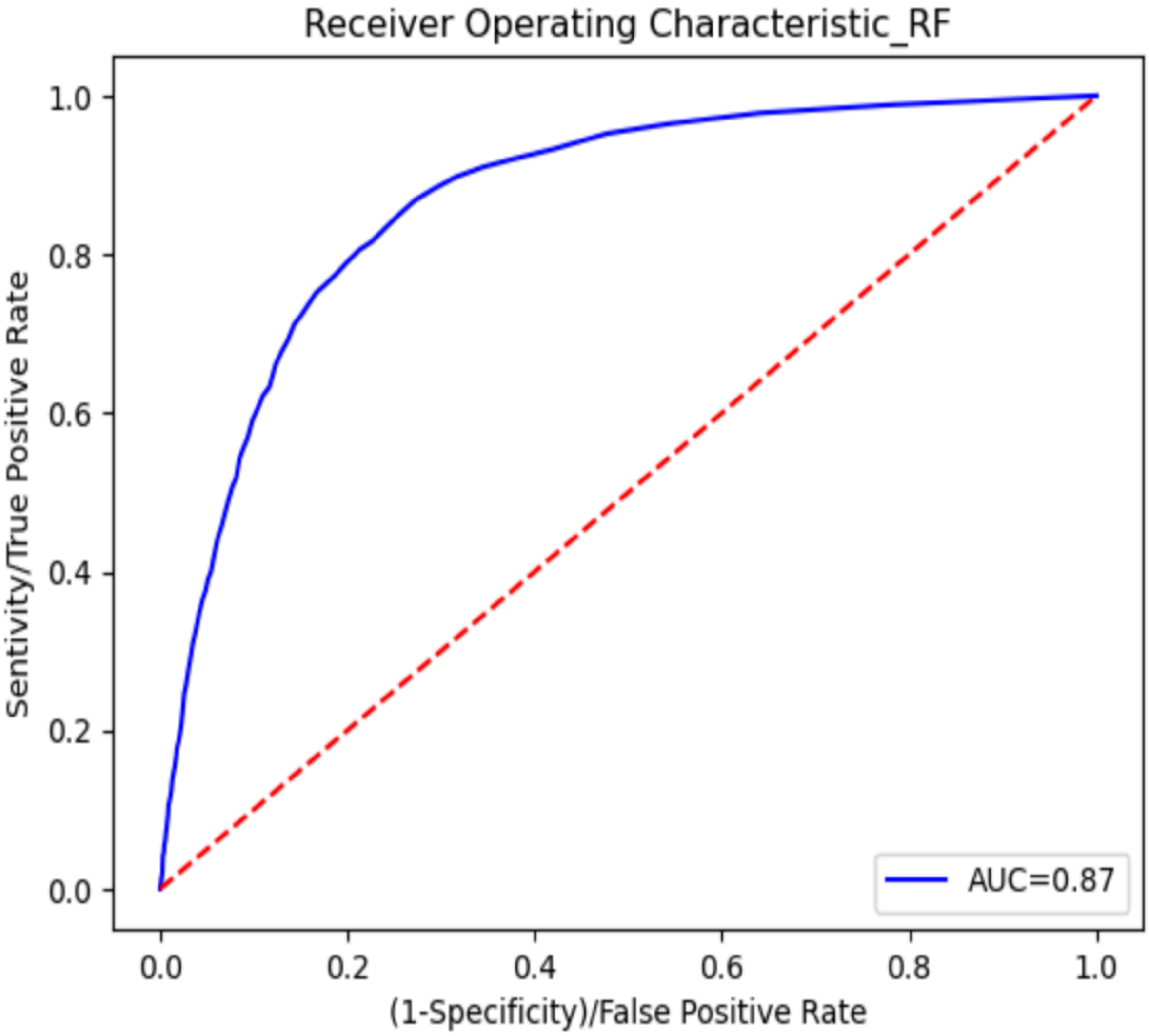


Figure 14 RF_ROC curve

2.3.2.3 Cumulative Gain Curve

The cumulative gain curve, shown in *Figure 15,* evaluates the performance of the Random Forest model in comparison to random selection. It illustrates the proportion of responders identified when targeting a certain percentage of the sample. The blue line represents the baseline, corresponding to random selection, while the orange line represents the response model. The curve demonstrates that by targeting just the top

20% of observations, the model captures over **60%** of responders. By targeting **80%** of the cases, the model successfully identifies all responders. This analysis underscores that the Random Forest classification model significantly outperforms random selection in efficiently identifying responders.

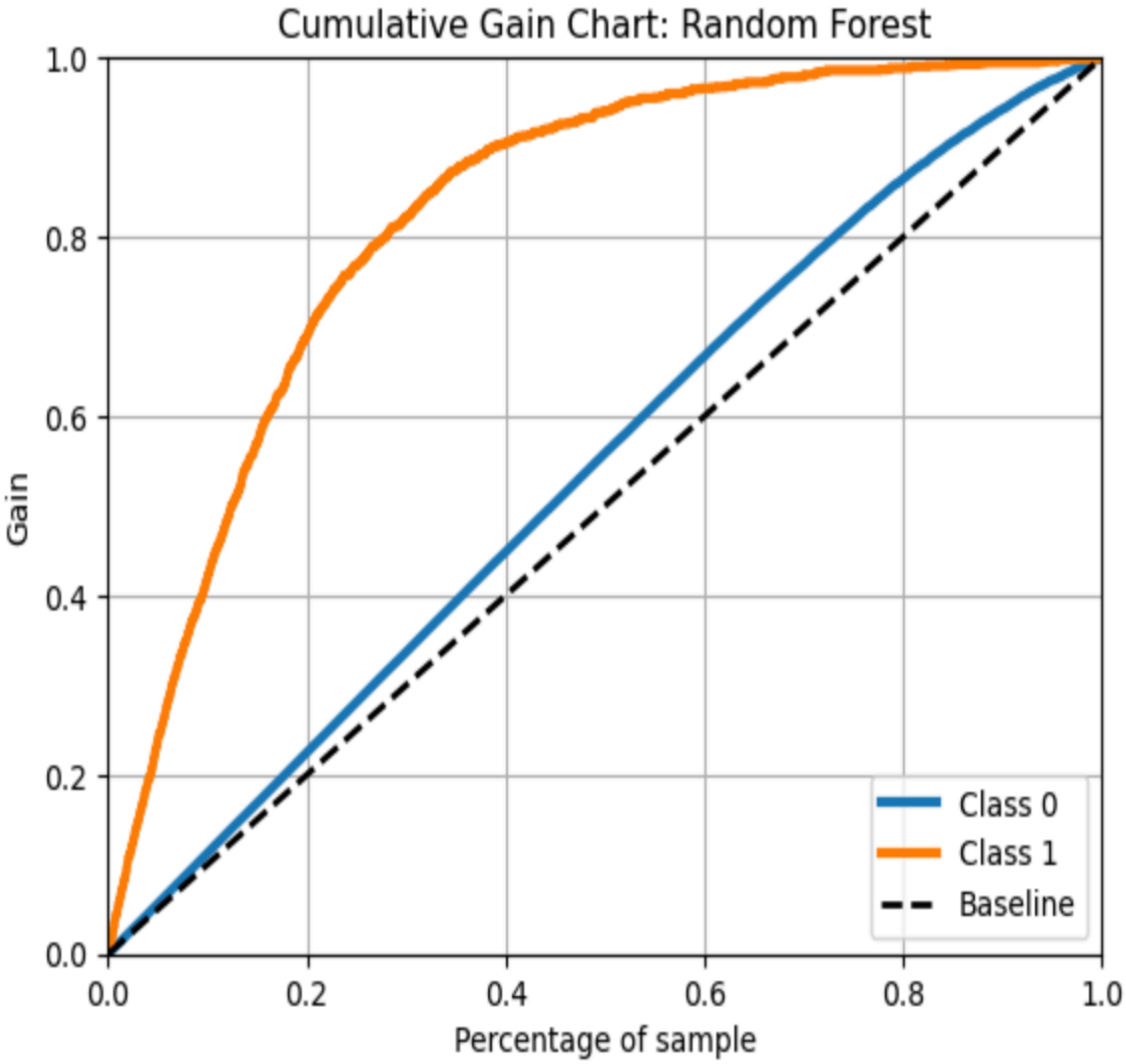


Figure 15 RF_cummulative gain curve

3 Conclusion

Metric	***Logistic Regression***	***Decision Tree***	***Random Forest***
AUC	87%	86%	87%
Accuracy	88.33%	87.55%	86.98%
Sensitivity	40.84%	44.87%	55.46%
Specificity	94.62%	93.20%	91.15%
F1- Score	45.02%	46%	49.9%
True Positives	486	534	660
False Positives	483	611	795
True Negatives	8500	8372	8188
False Negatives	704	656	530

Table 1 Summary

The **Random Forest model** emerges as the most favourable choice based on the metrics. Although Logistic Regression achieves the highest **accuracy (88.33%)** and **specificity (94.62%),** Random Forest excels in critical metrics like **Sensitivity (55.46%)** and **F1-score (49.9%),** making it more effective in identifying true responders. This higher sensitivity and F1-score indicate a better balance between precision and recall, which is crucial for applications like marketing campaigns that prioritize capturing positive responses. Additionally, Random Forest has the highest **True Positives (660)** and the lowest **False Negatives (530),** further demonstrating its robustness in correctly identifying responders. While the Decision Tree model shows similar performance, Random Forest offers a more balanced approach, making it the recommended model for maximizing campaign effectiveness and minimizing missed opportunities.