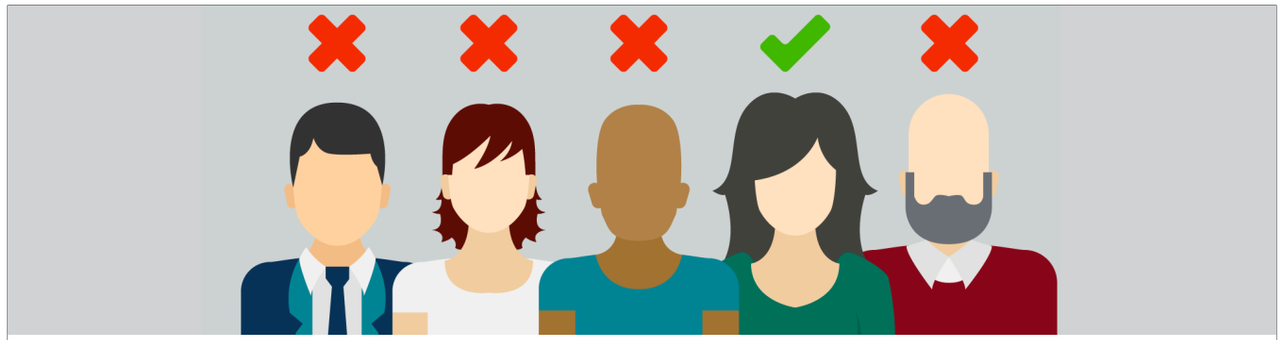


DATA-DRIVEN RECOMMENDATIONS

WHO TO CALL?

Good afternoon. For those that don’t know me, I’m Nadine a data scientist based in the London office. I’m here to present the results of a data science project where we supported the marketing team and helped increase revenue.

Image source: https://appsamurai.com/mobile-audience-targeting-how-to-hit-the-bulls-eye/

PROBLEM STATEMENT

- 2000 new customers
- Fixed term deposit telemarketing campaign
- Budget for 500 calls

Goal

- Identify which 500 customers to contact to maximise revenue
- Derive recommendations for future campaigns

The bank had just on boarded 2000 new customers and wanted to offer them a term deposit subscription like for existing customers. This was to be achieved through a telemarketing campaign. However due to budget and resourcing constraints, only 500 calls could be made.
The primary goal was to identify which of the 500 customers to contact to maximise revenue. We also wanted to provide recommendations for future campaigns driven by the data.

Image source: https://cmglocalsolutions.com/blog/know-your-target-audience-through-digital-tools

BUSINESS VALUE

- Successful subscription = revenue
  - $100 per subscription
  - Average uptake: 10-15%
  - Random 500 customers
  - Expected revenue: $5,000-7,500
- Use data science to increase revenue

A successful subscription to a term deposit translates directly into revenue for the bank, as it increases the bank's liquidity and fosters a stronger relationship with the customer. Through speaking with colleagues, we valued each subscription at $100. From past experience, we know uptake for such product through telemarketing campaign is low, at around 10-15%. So contacting the 500 customers randomly would give us expected revenue of between $5,000 and $7,500.

We used data science and predictive modelling to increase revenue.

Image source: https://lucrumconsulting.net/4-ways-to-increase-revenue/

METHODOLOGY

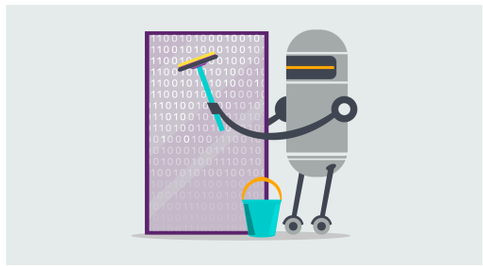
1. Gather and clean data 
2. Explore and visualise data 
3. Train and evaluate models 
4. Final predictions and recommendations 

To solve our problem we performed the following steps. In the next slides, we will go through each step one by one.

Icons source: from OSEMN framework originally by Hilary Mason and Chris Wiggins

1. GATHER AND CLEAN DATA

- Just over 39,000 customer data points
- 20 predictive features including
    - Personal attributes
    - Financial
    - Campaign
    - Economic indicators
- Clean: missing values, syntax

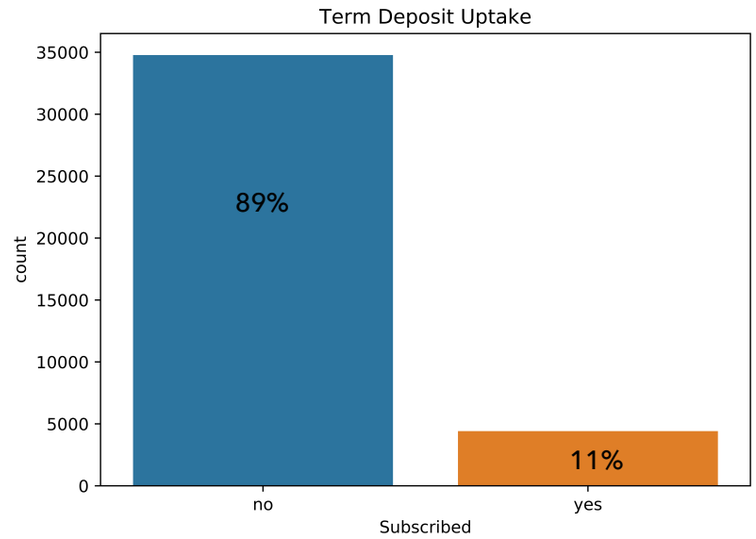


By looking at existing customer data, we had over 39,000 data points. We selected 20 features we thought would be worth investigating and help predict whether a customer would subscribe or not. These include personal attributes (job, age, education), financial (housing loan, personal loan, default credit), campaign (previous campaign results, means of contact, month of contact), economic indicators (consumer price index)

Cleaning involves preparing the data for modelling, such as addressing missing values and removing certain characters from the syntax.

Image source: https://lab.getapp.com/importance-of-data-cleaning-and-governance/

2. EXPLORE AND VISUALISE DATA



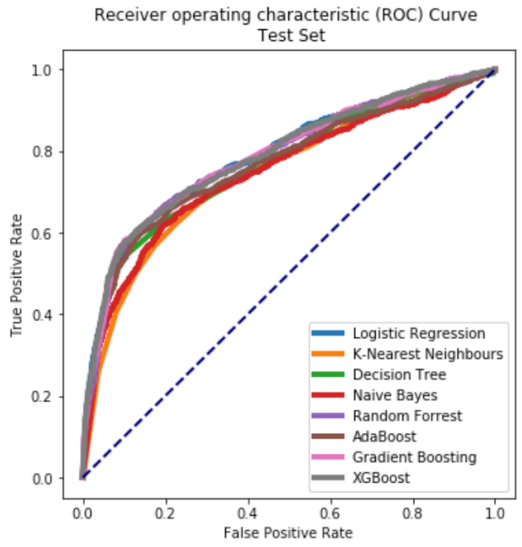
In this visualisation, we looked at the number of customers who subscribed to the term deposit in our pool of 39,000 observations. We see that for this campaign uptake was around 11% for our existing client base. This imbalance meant that we had to undertake certain steps to ensure our model was useful and be careful selecting our evaluation metric. I'd be happy to go into further details into the steps taken at the end, time permitting.

3. TRAIN AND EVALUATE MODELS

	Accuracy	F1	Precision	Recall	Profit
Logistic Regression	0.83	0.44	0.36	0.62	38980
KNN	0.60	0.3	0.19	0.75	33860
Decision Tree	0.86	0.47	0.42	0.54	36050
Naive Bayes	0.56	0.28	0.17	0.78	31510
Random Forrest	0.82	0.44	0.35	0.62	38630
Adaboost	0.85	0.46	0.40	0.57	36910
Gradient Boosting	0.85	0.47	0.45	0.59	38760
XGBoost	0.87	0.49	0.38	0.56	**39050**

With our data prepared, we tried various classification models to predict which customers would subscribe. The different algorithms are presented on the left-most column. We then looked at the model's performance across various metrics.
We defined a custom profit metric which took into consideration the expected revenue and cost of a call. I would be happy to go over details in the end and have details in the appendix. For this metric, the higher the value the better the model. As such our final chosen model was an XGBoost classifier. It is essentially a collection of decision trees, think flowcharts where each additional tree is focussed on correcting the errors of the previous one.

3. TRAIN AND EVALUATE MODELS



With our data prepared, we tried various classification models to predict which customers would subscribe. The different algorithms are presented on the left-most column. We then looked at the model's performance across various metrics.
We defined a custom profit metric which took into consideration the expected revenue and cost of a call. I would be happy to go over details in the end and have details in the appendix. For this metric, the higher the value the better the model. As such our final chosen model was an XGBoost classifier. It is essentially a collection of decision trees, think flowcharts where each additional tree is focussed on correcting the errors of the previous one.

4. FINAL PREDICTIONS

- 2000 new customers
- 500 calls, subscription value = $100
- Expected revenue: $5,000-7,500

- Used XGBoost classification model to select 500
- 144 customers subscribed
- **$14,400 revenue**

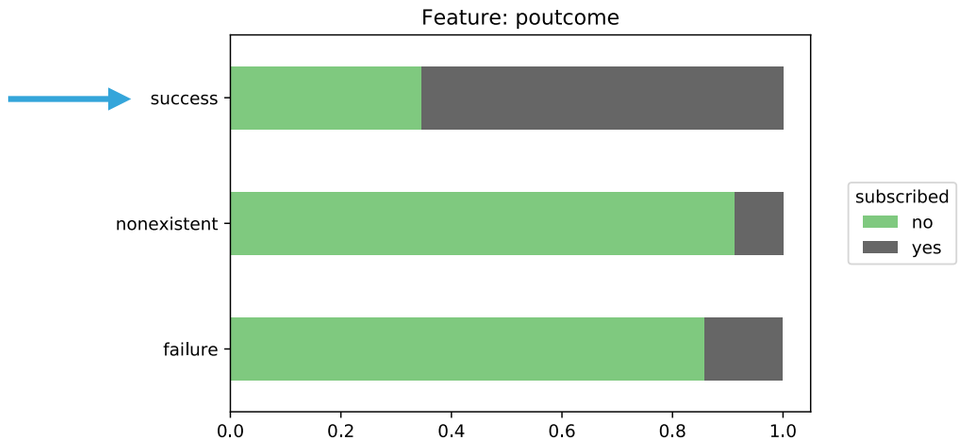
Recall our problem statement. We had 2000 new customers and were making 500 calls. With a subscription valued at $100 and uptake between 10-15%, the expected revenue was between $5,000 and $7,500.

Well we used our XGBoost classification model to select the 500 customers to call. Out of these 144 subscribed, almost 30%. This resulted in revenue of $14,400.

RECOMMENDATION - STUDENT/ RETIRED

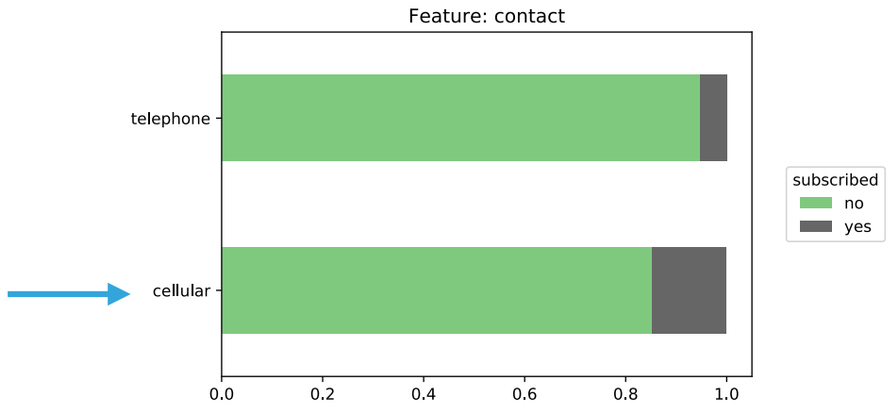
We investigated the rate of subscription amongst various job categories and the results are plotted in the form of a stacked bar chart. We can see that a greater proportion of students and retired customers said yes. As such we would recommend targeted campaigns for these types of customers.

RECOMMENDATION - PREVIOUSLY SUBSCRIBED



We looked at the results of the previous marketing campaign and noted that 65% of subscribers who previously subscribed to a term deposit (value success indicated by the blue arrow) subscribed this time round. As such keeping track of who is interested in the term deposit is useful and these customers should be contacted as a priority.

RECOMMENDATION - CELLULAR CONTACT



We investigated how a customer was contacted and noted a difference between landline and cellular. It appears that cellular calls had a higher chance of resulting in a subscription and thus should be the preferred method of contact.

FUTURE WORK

- Customer segmentation
- Determine optimal number of calls
- Seasonality

Customer segmentation: We would use unsupervised machine learning techniques, namely clustering, to gain a better understanding of the bank's customer base. This would help not only target future similar campaigns but also improve the bank's overall relationship with customers.
Determine optimal number of calls: In this scenario, we were informed that the bank had the budget for 500 calls. But what is the actual optimal number of calls to ensure most potential subscribers are reached, whilst not wasting resources?
Seasonality: We would establish the best month for launching the next campaign, to maximise success.

THANK YOU

NADINE AMERSI-BELTON

nzamersi@gmail.com
datascimum
nadinezab

I hope this presentation has shown you the value gained from applying data science tools to the marketing campaign and welcome any questions you may have.

PROFIT METRIC 1/2

	Predict No	Predict Yes
Actual No	TN	FP
Actual Yes	FN	TP

We begin with a confusion matrix, which shows all possible outcomes.
In our scenario, a **false negative** occurs when the model predicts that a customer will not subscribe (target `no`) when in fact they would have. This is hugely detrimental, as bank XYZ would lose out on potential revenue. We estimate the revenue resulting from a subscription be valued at USD 100. As such we need to ensure the false negatives predicted by our model are minimised.
On the other hand, a **false positive** occurs when the model predicts that a customer will subscribe (target `yes`) when in fact they won't. In this case, we have wasted the cost of a telephone call.
As such, **false negatives are worse than false positives**.

PROFIT METRIC 1/2

	Predict No	Predict Yes
Actual No	0	- C (False Positive FP)
Actual Yes	0	R - C (True Positive TP)

Profit = TP * (R-C) + FP * (-C)

To create a metric specific to our problem, let us introduce a profit matrix, assigning values to possible outcomes.
A false positive has a cost associated to it, as we have wasted a call. This is denoted by -C. On the other hand a true positive results in revenue less the call cost, denoted by R-C.
We can then define our profit metric by the formula on the slide.

PROFIT METRIC 2/2

	Predict No	Predict Yes
Actual No	0	-10
Actual Yes	0	100 - 10

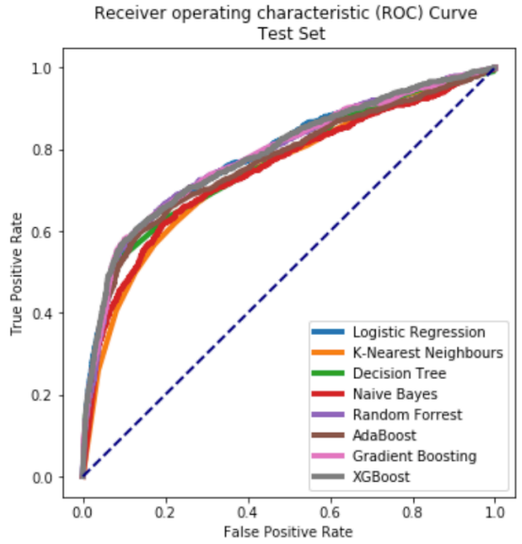
False Positive FP

True Positive TP

Profit = 90TP - 10FP

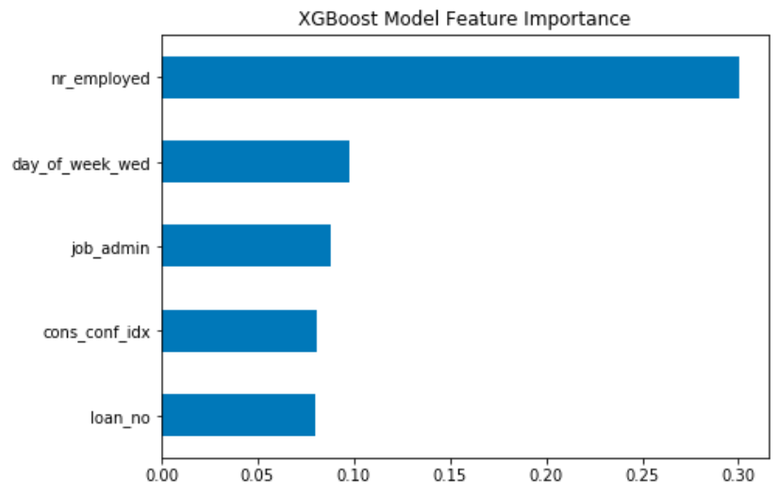
Based on domain knowledge, we assign values to C and R. In this project, we have been informed that stakeholders value a subscription at USD 100. Similarly, the telemarketing calls have been assigned a cost of USD 10 per call.

TRAIN AND EVALUATE MODELS - ROC



Receiver Operating Characteristics (ROC) curve and associated AUC (Area under Curve) metric
The ROC curve is created by plotting the true positive rate (TPR) against the false positive rate (FPR) at various threshold settings.
The dotted line represents a no skill model which guesses, i.e. 50% chance of being accurate.
We see that all classifiers perform similarly and way above the no skill benchmark.

FEATURE IMPORTANCE



The 5 features which had the highest impact on the classification are:
1. number of employees (a quarterly metric which represents economic state)
2. Whether contacted on a Wednesday
3. whether the customer's job falls into the admin category
4. Consumer confidence index
5. whether the customer has a personal loan

LOGISTIC REGRESSION

- Uses logistic function
- Best parameters:
  - C = 0.1
  - Solver = Liblinear
- Accuracy: 0.83
- F1: 0.44
- Recall: 0.63
- Profit: 38980

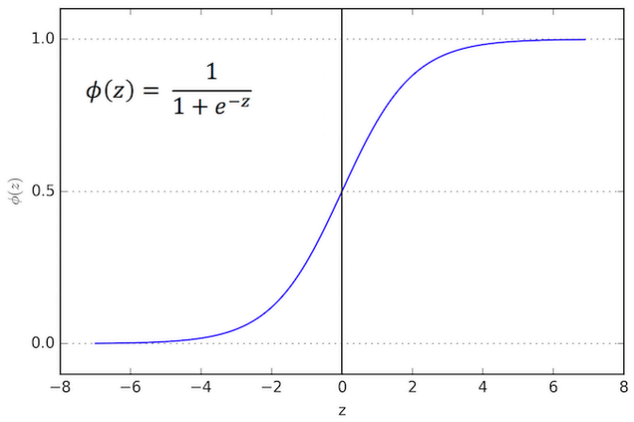


Image: https://www.analyticsvidhya.com/blog/2020/04/machine-learning-using-c-linear-logistic-regression/logistic-4/

K-NEAREST NEIGHBOURS

- Distance-based classifier
- Best parameters:
  - Number of neighbours = 15
  - Weights = uniform
  - P = 5 (Minkowski power)
- Accuracy: 0.60
- F1: 0.30
- Recall: 0.75
- Profit: 33860

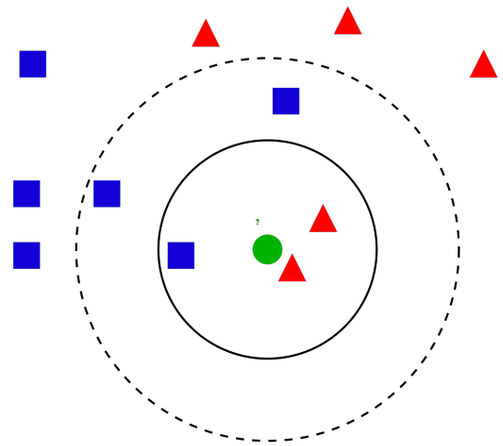
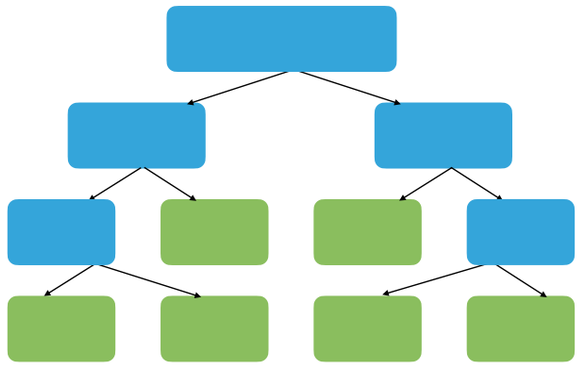


Image source: https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/

DECISION TREE

- Directed acyclic graph, greedy algorithm
- Best parameters:
  - Criterion = Gini
  - Max Depth = 5
  - Min Samples Split = 3
- Accuracy: 0.84
- F1: 0.31
- Recall: 0.32
- Profit: 20280



NAIVE BAYES

- Based on Bayes' Theorem
- Naive assumption of independence between features
- Accuracy: 0.56
- F1: 0.28
- Recall: 0.78
- Profit: 31510

Image source: https://en.wikipedia.org/wiki/Thomas_Bayes

RANDOM FORREST

- Ensemble method, multiple decision trees
- Best parameters:
  - Criterion = Gini
  - Number of estimators = 100
  - Max depth = 4
  - Min samples split = 10
- Accuracy: 0.82
- F1: 0.44
- Recall: 0.62
- Profit: 38630

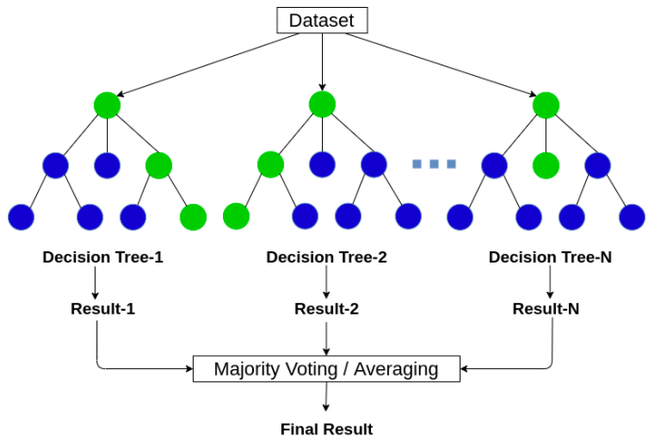


Image source: https://www.analyticsvidhya.com/blog/2020/05/decision-tree-vs-random-forest-algorithm/

ADABOOST

- Ensemble method
- Adjusts weights to focus on difficult cases
- Best parameters:
  - Number of estimators = 100
  - Learning rate = 1
- Accuracy: 0.85
- F1: 0.46
- Recall: 0.57
- Profit: 36910

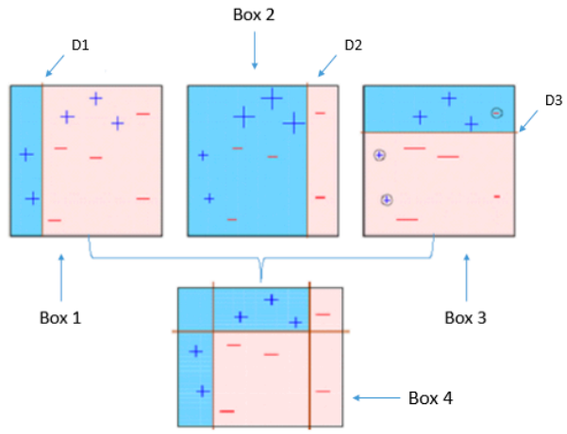


Image source: https://towardsdatascience.com/understanding-adaboost-2f94f22d5bfe

GRADIENT BOOSTING

- Ensemble of weak learners, decision trees
- Optimizes differentiable loss function
- Best parameters:
  - Number of estimators = 20
  - Max depth = 3
  - Min samples split = 3
- Accuracy: 0.85
- F1: 0.47
- Recall: 0.59
- Profit: 38760

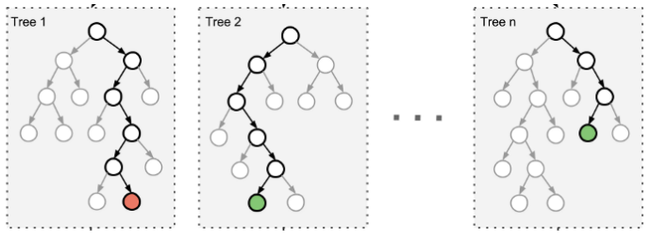


Image source: https://blog.statsbot.co/ensemble-learning-d1dcd548e936

XGBOOST

- Ensemble method, multiple decision trees
- Best parameters:
  - Number of estimators = 20
  - Max depth = 3
  - Min child weight = 3
- Accuracy: 0.87
- F1: 0.49
- Recall: 0.56
- Profit: 39050