The American Express Campus Challenge 2024

Raman Kumar
Sachin Kumar
Prabhash Mishra
IIT KANPUR

Dikec

Summary

Final Strategy*
((merchant1 !='163070' and '164428') and merchant2='163070') or (merchant2='0' and (merchant1!='0' and '163070' and '164428'))) Or ((merchant1='0' and merchant2='0') and addr_mismatch='Y') or (25<=risk_score_3) and (risk_score_4=0) and (addr_mismatch != 'NA' and (merchant1=['163070' , '164428'])')

** Final Strategy = The strategy used to get the final score on the leaderboard*

Derived Variables			
#	**Variable Name**	**Derivation Logic**	**Description**
1	none	none	A small description that gives details about this variable
2	none	none	A small description that gives details about this variable
3	….	….	….
4	….	….	….

Story Behind the Strategy

- We began the fraud detection task by thoroughly analyzing all the variables and understanding how each can be utilized in the fraud detection process.
- Most machine learning models used for fraud detection, such as logistic regression and random forests, generate a probability score for each transaction, representing the likelihood of fraud. In our dataset, we used the variable **risk_score_3** to capture this probability as it was Risk score assigned based on probability of fraud.
- Now, we had to correctly tune a threshold for this variable. Neither a high threshold nor a low threshold. As,
    1. A **higher threshold** increases hit rate but reduces coverage (fewer transactions are flagged as fraud, but those that are flagged are more likely to be correct).
    2. A **lower threshold** increases coverage but reduces hit rate (more transactions are flagged as fraud, but there are more false positives).
- We initially set the threshold by calculating the average value of **risk_score_3** across all default indicators(=39), as it provided a balanced starting point—neither too high nor too low.
- We flagged any value above the threshold of **risk_score_3** as fraud, starting with a threshold of >39, which identified 98 fraudsters among the defaulters. Through iterative adjustments, we refined the threshold and determined the final value to be **≥25** for optimal fraud detection.
- Now that we have identified **risk_score_3** as one key variable, our next step is to explore additional variables to improve the hit rate and coverage. The goal is to identify fraudsters across the entire dataset without relying on default indicators, using them merely as a hint.
- We plotted different variables against **risk_score_3** to identify potential relationships of only the defaulters. This helped us discover relevant variables linked to fraud. Based on the insights, we determined threshold values for those variables to enhance detection and then applied to the whole dataset.

- Other variables and their thresholds are as follows

  1. Merchant 2. Income between 50000 to 85000

  3. acq_channel='Channel 3' 4. prod_name='Product 1' 5. payments<=4300

- After calculating the thresholds for various variables, we selected five—**risk_score_3**, **risk_score_4**, **risk_score_11**, **risk_score_1** and **payments**—for our strategy. We then applied an "AND" condition across these thresholds to evaluate the entire dataset for potential fraud.
- Also, there were few false positives in our strategy but there was no true negative. The unique_identifier of fraudsters caught by our strategy which were defaulter (true positive) had unique_identifier in a serial number which was also hinted in the problem statement (surge) and was helpful in finding fraudsters.

We have mentioned an alternate strategy different from our final strategy

a) Mention in detail the approach behind that strategy

The approach for both strategies was quite similar; we started and proceeded in the same manner, calculating thresholds consistently. The key difference was the identification of unique identifiers for **merchant 1** and **merchant 2** from the list of true positives. By applying these identifiers, we aimed to improve our scores, leading to our final strategy.

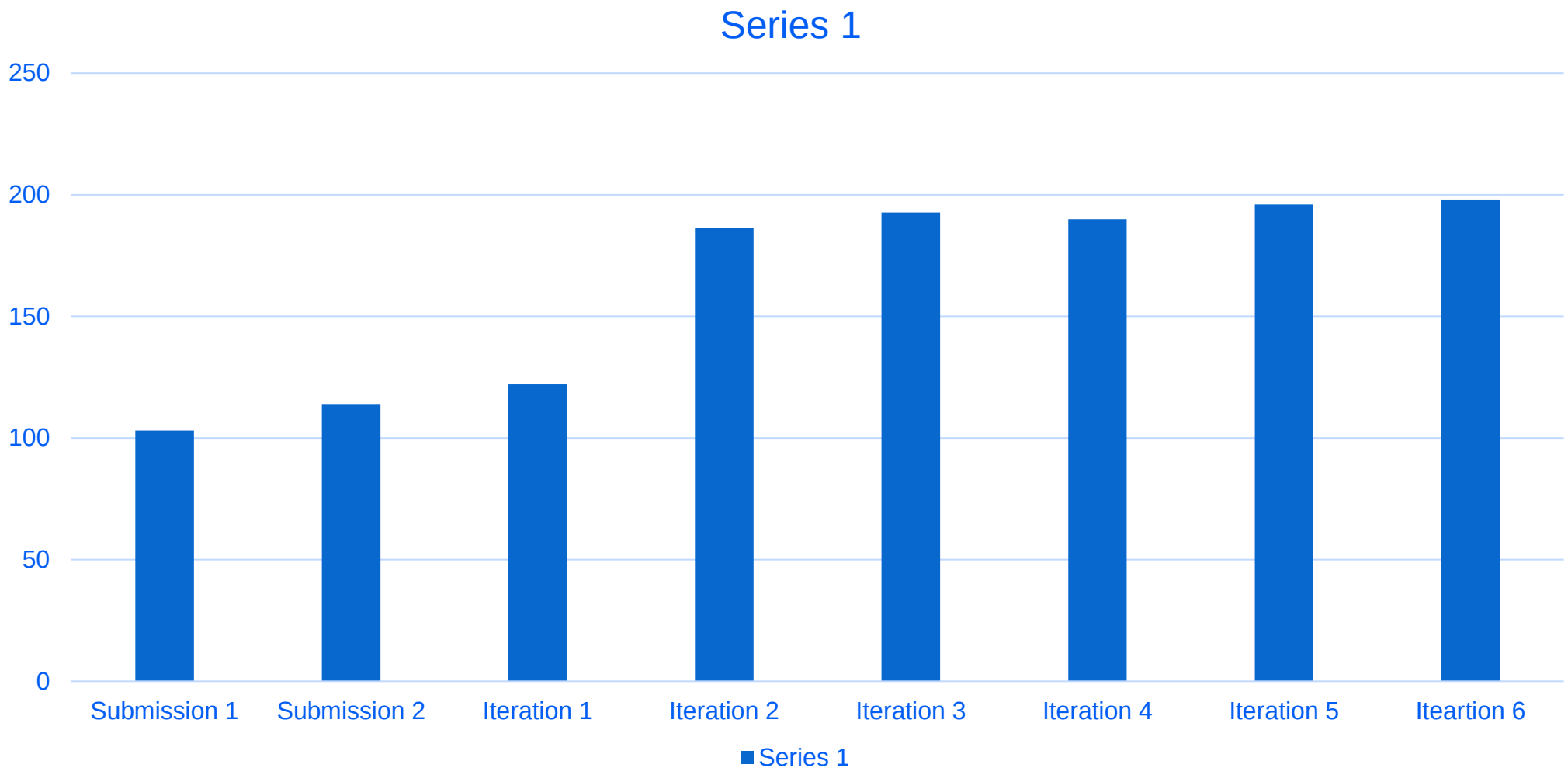
b) Why do you think this strategy is more generalized than your original strategy?

In the final strategy, we didn't apply any thresholds for **merchant 1** or **merchant 2**; instead, we utilized the unique identifiers associated with them. While this approach can be effective, it is inherently dynamic and resembles a hard-coded solution. In contrast, our overall strategy employs thresholds from various variables to flag potential fraudsters. This means that if someone attempts to exploit weaknesses in the acquisition system, they are more likely to be detected by our strategy, ensuring a more robust fraud detection process.

Iterative Progress

- **Submission 1 and 2** : In our initial submission, we primarily aimed to grasp the question but didn't fully understand it. The prompt requested a strategy that didn't involve a modeling solution, yet we approached it using a linear SVC method, Random forest and Isolation Forest respectively. This misalignment highlighted our need to focus more on strategy formulation rather than relying on modeling techniques to address the problem effectively. From next submissions we understood the task of not using modelling based rather focused on formulating a strategy.
- **Iteration 1/ Submission 3** : In the first actual iteration, where we used a strategic approach instead of a modeling solution, we applied the average value of **risk_score_3**, which was **>39**. Additionally, we noticed a cluster forming at **risk_score_4 = 0**, so we incorporated these threshold values into the strategy. We then applied this combined threshold to the entire dataset for fraud detection. Using this strategy, we detected 207 fraud cases across the entire dataset, achieving a better score than our previous submissions. This improvement demonstrated the effectiveness of applying targeted thresholds for fraud detection.
- **Iteration 2/ Submission 4** : We needed to use more variables to eliminate all those false positives and needed to lower the value of **risk_score_3** to increase the hit rate. We decreased the threshold value of **risk_score_3** from >39 to >=25 just to increase the number of defaulters satisfying this threshold from 98 to 125. By carefully analyzing clusters of points in individual plots and intersecting them across different plots, we identified unique identifiers that met the cluster conditions (or the specific threshold for the cluster of points). By analyzing all these defaulter data points, we came with variables to be used and their thresholds.
  - Strategy of the Iteration : (risk_score_4=0) and (0.1<=risk_score_1<=5.8) and (risk_score_3>=25) and (50000<=income<=85000) and (payments<=3107)
  - This strategy gave a decent score having only few false positive and true negative. Hence the threshold value were good.

- **Iteration 3/ Submission 5** : In previous iterations, the variables used had well-optimized thresholds. In this iteration, the focus is on improving our score by incorporating additional valuable variables, such as **Merchant 1**. This involves analyzing its relationship with fraudsters and identifying the **Merchant 1** associated with fraudulent activities in the dataset. The goal is to leverage these variables to achieve a hit rate of 1, effectively identifying all fraudsters within the dataset. Strategy used (risk_score_4=0) and (0.1<=risk_score_1<=5.8) and (risk_score_3>=25) and (50000<=income<=85000) and (merchant1=163070 or merchant1=167574 or merchant1=110104 or merchant1=166949 or merchant1=164435 or mechant1=164428 or merchant1=164845 or merchant1=164428 or merchant1=0)
- **Iteration 4/ Submission 6** : In this iteration, we implemented the strategy discussed in the previous strategy review. While the overall score of this strategy is lower compared to the previous one, the variables used have better-optimized thresholds. This is the strategy we discussed in the strategy story. Using **Merchant 1** requires accounting for all merchants in the system, which makes it an unreliable variable due to its highly dynamic nature. Its variability reduces its effectiveness in identifying fraud consistently, making it less suitable for the strategy. Strategy of the iteration (risk_score_4=0) and (0.1<=risk_score_1<=5.8) and (risk_score_3>=25) and (risk_score_11>=0.2787) and (payments<=4300)
- **Iteration 5/ Submission 7 :** Same as the iteration 5, we use merchant 1 together with suitable variables which go along with merchant 1 to determine the frauds in the dataset. From the list of fraudsters already determined by other strategies we look out for variable which account for a smaller number of false positive giving as a higher score for this vary iteration. The variable which accounts for this is addr_mismatch. Strategy of the iteration ((merchant1 !='163070' and '164428') and merchant2='163070') or (merchant2='0' and (merchant1!='0' and '163070' and '164428')))Or ((merchant1='0' and merchant2='0') and addr_mismatch='Y') or (25<=risk_score_3) and (risk_score_4=0) and (addr_mismatch != 'NA' and (merchant1=['163070' , '164428'])')
- **Iteration 6/ Submission 8 :** This is the final strategy which gave as the highest score. This strategy is very similar to the iteration 6 which used the same merchant 1 and addr_mismatch. Just it does not count for a single false positive which is missed by this iteration for the given data.


Series 1
250
200
150
100
50
0
Submission 1
Submission 2
Iteration 1
Iteration 2
Iteration 3
Iteration 4
Iteration 5
Iteartion 6
Series 1

Other Recommendations

1. How do you think your strategy (and/or alternate strategy) can be scaled?

   Scaling a threshold-based fraud detection strategy involves several key enhancements to improve its effectiveness and adaptability. First, integrating diverse data sources, such as transaction history and external databases, enriches the dataset, allowing for more accurate threshold settings. Regularly retraining machine learning models with updated data ensures they remain responsive to evolving fraud patterns. Implementing automated alert systems for high-risk transactions streamlines the detection process and improves response times. Additionally, adopting dynamic thresholds based on real-time analysis allows for adjustments according to current risk levels, while continuous feedback from investigations helps refine thresholds further.

2. Share your opinion on how your strategy might fare against similar fraudsters in future?

   The fraudsters in this vary dataset produced a new acquisition system for doing frauds. If similar ways are followed by fraudsters in the future for doing frauds. It will surely be caught by this strategy with a minimal rate of false alarm.

3. What are some of the challenges you faced in coming up with a strategy?

   Choosing appropriate variables , and then cluster from data and define the threshold of each cluster or the variable so that fraudsters can be caught and understanding the meaning of all these variables provided and act accordingly.

4. Please share additional details if you think any other intelligence could be leveraged to improve your strategy’s performance

   1. Machine Learning Model Optimization
   2. Automation and Real-Time Processing

Appendix

https://colab.research.google.com/drive/1yCt74dMDuMiPUwHk2RzvXLEgq2Ae-6zX?usp=sharing

https://docs.google.com/document/d/1ufAzhzqOVfVvS_5_m0ZAFRPdtZ0GQWS00_63dIsc4yo/edit?usp=sharing