GENROCKET SYNTHETIC TEST DATA FOR MACHINE LEARNING IN FINANCIAL SERVICES

Highlights

- **Industry**
  - A major financial services company (credit card data)
- **GenRocket Products**
  - Synthetic data automation
- **Challenges**
  - Large volume of data needed for RFM modeling (recency, frequency, monetary value)
  - Reliance on production data required a lengthy approval process, translating to a lead time of 2-4 weeks.
  - Relying on data from the production environment required significant manual work to customize and format data to fit different test cases, leading to just 15% test coverage.
- **Solution**
  - The company's data and analytics team created a complex scenario chain, consisting of 24 domains and 24 scenarios, for generating synthetic data representing 1 million grocery store transactions.

Summary

Supermarket and grocery store transactions generate an enormous volume of data from customer purchases, making it ideal for RFM modeling. However, privacy restrictions prohibit using data verbatim. This financial services company needed a faster way to create production data to fit various test case scenarios for a new machine learning/AI project.

About the Client

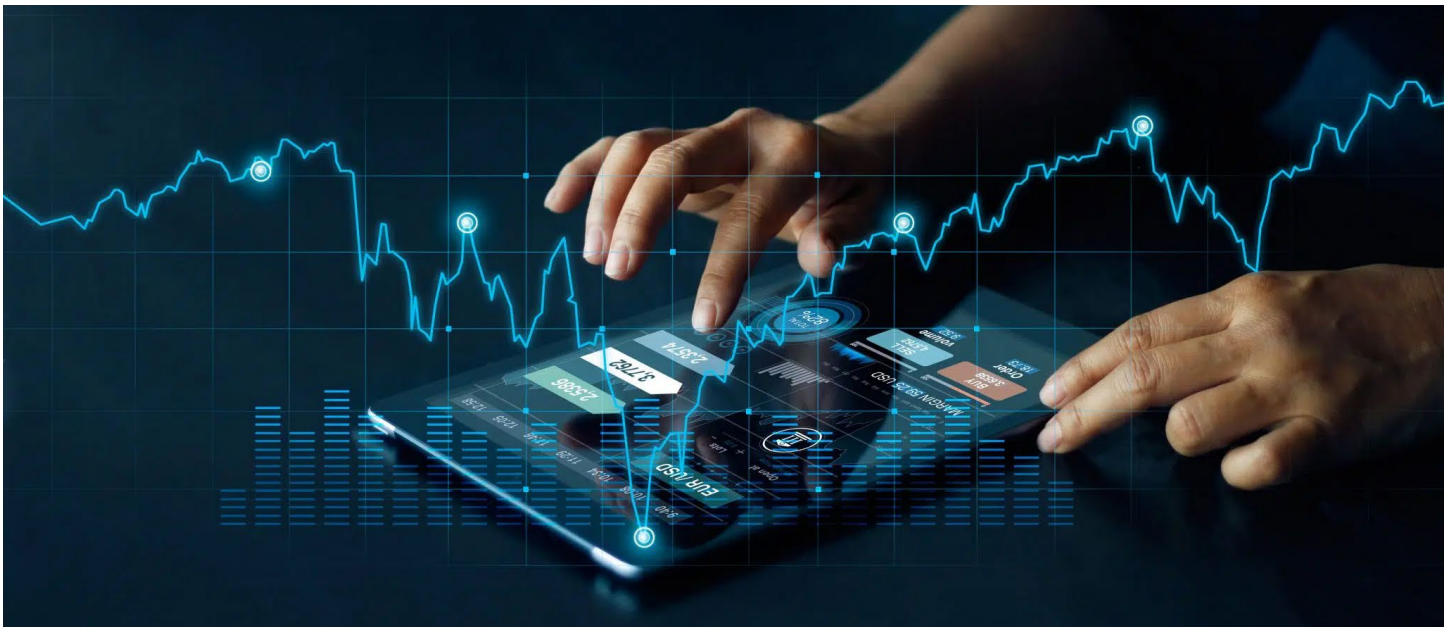
The client is a large financial services company; their credit card division was instrumental in this project.

The Problem

The financial services sector has realized countless use cases for AI, from identifying instances of fraud to assisting with customer-facing applications. Predictive modeling has proven to be a valuable use case for AI, where machine learning algorithms can pick up on small nuances in transactions and the context around them to predict future purchases. Machine learning algorithms require huge amounts of data to be trained effectively, and the sensitive nature of financial transaction information can make that data difficult and time consuming to obtain.

That's why one major financial services company turned to GenRocket when it needed robust testing data for training a prototype machine learning algorithm designed to identify cross-selling opportunities between grocery stores and restaurants. For the algorithm to effectively identify patterns in the data, it needed access to massive volumes of grocery store transactions, which privacy rules make difficult to access and leverage. As a result, the company's data and analytics teams faced a couple key hurdles as they built their prototype model:

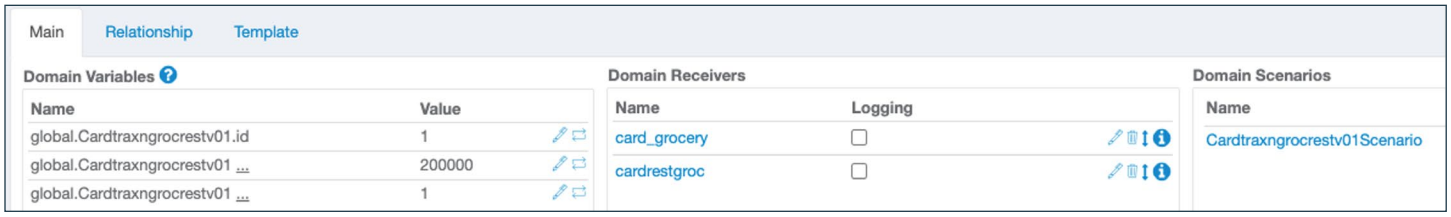
- Reliance on production data required a lengthy approval process, translating to a lead time of 2-4 weeks.
- Relying on data from the production environment required significant manual work to customize and format data to fit different test cases, leading to just 15% test coverage.



The Solution

The company's data and analytics team used the GenRocket Test Data Automation platform to create a complex scenario chain, consisting of 24 domains and 24 scenarios, for generating synthetic data representing 1 million grocery store transactions.

Synthetic data from GenRocket was then fed into Amazon Data Wrangler, where the RFM (recency, frequency, monetary) model would create prediction scores, identifying customers that would likely use the same credit card for restaurant purchases.



By eliminating the need to rely on production data, testers no longer needed to go through the lengthy approval process and could generate training data with similar statistical distribution on demand. With robust realistic synthetic data available at their fingertips, testing teams also achieved far greater test coverage in less time than it took to get approval for using production data.

Outcomes

- **Eliminated the lead time** required to leverage data from the production environment
- Reduced cycle time from **two days to 30 minutes**
- Increased regression test coverage **from 15% to 50%**

The company was also able to build out sets of re-usable test data and business logic, which allowed testing teams to build more complex scenarios and scenario chains for testing other tools and software. Each business unit assigned an internal subject matter expert to help individual users build their own scenarios and scenario chains, activating the benefits of GenRocket's self-service model. This allowed other business units to significantly reduce the time it takes to test new features and software while improving coverage for multiple kinds of testing.

The table below shows just a few of the benefits of scaling GenRocket across the organization:

Business Unit	GenRocket Benefits
Payments	• 1,400-hour reduction in cycle time • Increased systems integration testing and performance coverage from 30% to 80%.
Banking	• 388-hour reduction in cycle time • Increase in regression and API performance coverage from 0% to 80%
Credit Card	• Saved 1,300 hours during the first nine months of deployment • Increased regression coverage from 0% to 50%
Data and Analytics	• Reduced cycle time by more than 250 hours. • Improved component testing coverage from 0% to 50%

Over time, the credit card company plans to scale GenRocket to more than 100 teams to yield these benefits across the enterprise. As more of the organization builds test cases and scenarios, they'll have access to a wide variety of re-usable test data generation capabilities and business logic for testing by enabling QA team members to leverage synthetic data.

Conclusion

This company's data needs posed considerable challenges to the team manually producing data for testing. GenRocket's ability to quickly create synthetic test data according to various use cases and complexities resulted in considerable time saving without sacrificing accuracy and reliability.