Week 13 - Bring it all together

Maria Alice Fagundes Vieira

Aritra Ray

---

1. Problem Statement

Santander Bank's recommendation system currently suffers from inefficiencies, including irrelevant recommendations and overlooked customer needs. These issues lead to low customer engagement and lost revenue opportunities.

The primary goal of this project is to build a machine learning-based personalized product recommendation system that predicts the likelihood of customers purchasing specific financial products. This system should ensure that:

1. Customers receive relevant product suggestions based on their unique profiles and needs.
2. Santander can improve product adoption rates by focusing marketing efforts on the most likely prospects.

---

2. Articulation of Value

Customer-Centric Growth

Providing tailored product recommendations enhances customer satisfaction and loyalty. Instead of generic offers, customers receive suggestions aligned with their needs, increasing trust in Santander's services.

Revenue Growth

By improving the targeting mechanism, the bank can increase its conversion rate for product adoption from the current 5% to 15%. This translates into an additional $20 million in annual revenue.

Operational Efficiency

Automation of the recommendation process reduces dependency on manual customer profiling and sales interventions. With a scalable recommendation system, Santander can optimize its marketing budget and save labor hours.

Strategic Insights

A robust recommendation system also serves as a tool for better understanding customer behavior and preferences. Insights from this system can guide the design and development of new financial products.

3. Dataset Overview

Our dataset contained 47 columns, encompassing features like income, age, customer_code, and a range of product indicators (e.g., savings_acct, current_acct, credit_card, etc.). We began by cleaning the data to handle null values, remove duplicates, and ensure consistency in variable formats. Specific steps included:

Handling Missing Data: Imputed missing values in income using median imputation and replaced null entries in categorical variables like join_channel with the most frequent category.

Outlier Treatment: Identified and capped outliers in income and age using the interquartile range (IQR).

Feature Engineering: Created new variables such as income_to_age (income normalized by age) and income_to_product_ratio (income divided by the number of subscribed products) to enrich the dataset.

4. Methodology

4.1. Data Preprocessing

Handling missing data and inconsistencies was crucial to prepare the dataset for model training.

- **Missing Values:**
  - Variables like income and age had missing entries, which were imputed using median values to minimize bias.
- **Outliers:**
  - Extreme values, such as unusually high income values, were treated using interquartile range (IQR) thresholds.
- **Standardization:**
  - Continuous features like income and age were scaled to ensure uniformity and prevent model bias toward features with larger ranges.

4.2. Feature Engineering

To improve model performance, derived features were added:

1. **Income-to-Product Ratio:** A proxy for identifying customers with underutilized financial capacity.
2. **Income-to-Age Ratio:** A measure of financial stability, capturing the relationship between income and life stage.

3. **Total Savings Products:** The sum of binary values for savings-related accounts (e.g., savings, payroll, and deposits).

4.3. Dimensionality Reduction

While the dataset features were not highly correlated, we conducted feature importance analysis using a Random Forest Classifier. This step identified which features significantly impacted product recommendations and allowed us to drop irrelevant ones, reducing model complexity.

---

5. Model Development

5.1. Baseline Model

We began with a logistic regression model to establish a baseline for comparison. This simple model provided insight into the overall structure of the data and allowed us to validate the preprocessing pipeline.

5.2. Advanced Modeling

Subsequently, we experimented with more sophisticated models, including:

1. **Gradient Boosting Machines (GBM):** XGBoost and LightGBM were evaluated for their ability to handle imbalanced data and capture complex interactions.
2. **Neural Networks:** Tested for their potential to capture non-linear relationships and multi-class predictions.
3. **Ensemble Models:** Combined the strengths of GBM and Random Forest through stacking to improve performance.

5.3. Hyperparameter Tuning

For Random Forest, the following parameters were optimized using **GridSearchCV**:

- n_estimators: Values of 100, 200, and 500 were tested, with 200 providing the best trade-off between performance and computational efficiency.
- max_depth: Tuning revealed that a depth of 10 was optimal for avoiding overfitting.
- min_samples_split and min_samples_leaf: The best combination was found to be 4 and 2, respectively.

For Gradient Boosting, the optimized parameters were:

- learning_rate: A value of 0.05 yielded the best results after testing 0.01, 0.05, and 0.1.
- n_estimators: A setting of 300 provided the best balance of training time and model accuracy.
- subsample: A value of 0.8 minimized overfitting while maintaining model robustness.

5.4. Evaluation Metrics

Given the multi-class, imbalanced nature of the dataset, we prioritized the following metrics:

- **Mean Average Precision (MAP):** Measures ranking effectiveness, emphasizing relevance in top positions.
- **F1 Score:** Balances precision and recall to evaluate the classifier's overall effectiveness.
- **AUC-ROC:** Provides a measure of separability for binary classification tasks.

---

6. Economic Value Assessment

Assumptions:

1. Santander's current customer base: 1,000,000 customers.
2. Conversion rate increase: From 5% to 15%.
3. Average revenue per additional product sold: $200.
4. Model's annual operational effectiveness.

Value Derivation:

Current Revenue:

$R_{current} = 1{,}000{,}000 \times 5\% \times \$200 = \$10{,}000{,}000$

Projected Revenue:

$R_{projected} = 1{,}000{,}000 \times 15\% \times \$200 = \$30{,}000{,}000$

Incremental Value:

$R_{increment} = R_{projected} - R_{current} = \$20{,}000{,}000$

This $20 million annual revenue boost is conservative, excluding potential long-term benefits such as increased customer retention.

---

7. Results

7.1. Model Performance

The ensemble model emerged as the best-performing model, achieving:

- Mean Average Precision (MAP): 0.45.
- F1 Score: 0.58.
- AUC-ROC: 0.81.

7.2. Feature Insights

The most influential features for product prediction included:

1. **Income-to-Product Ratio:** High ratios correlated strongly with credit card and investment product adoption.
2. **Tenure:** Longer-tenured customers were more likely to adopt savings accounts and pensions.
3. **Customer Segment:** Top-tier customers showed higher interest in high-value products like loans and investments.

7.3. Practical Implementation

The model recommends the top three most probable products for each customer. A targeted marketing campaign can focus on customers with the highest likelihood of conversion.

---

8. Conclusion

This project demonstrates the value of machine learning in transforming Santander's product recommendation process. By leveraging customer data and predictive analytics, the bank can deliver tailored suggestions, improving customer experience and driving significant revenue growth. The final model offers actionable insights for marketing campaigns and customer engagement, with an annual revenue boost potential of $20 million. Further refinements, including real-time updates and continuous learning, can make the system even more robust and adaptive.