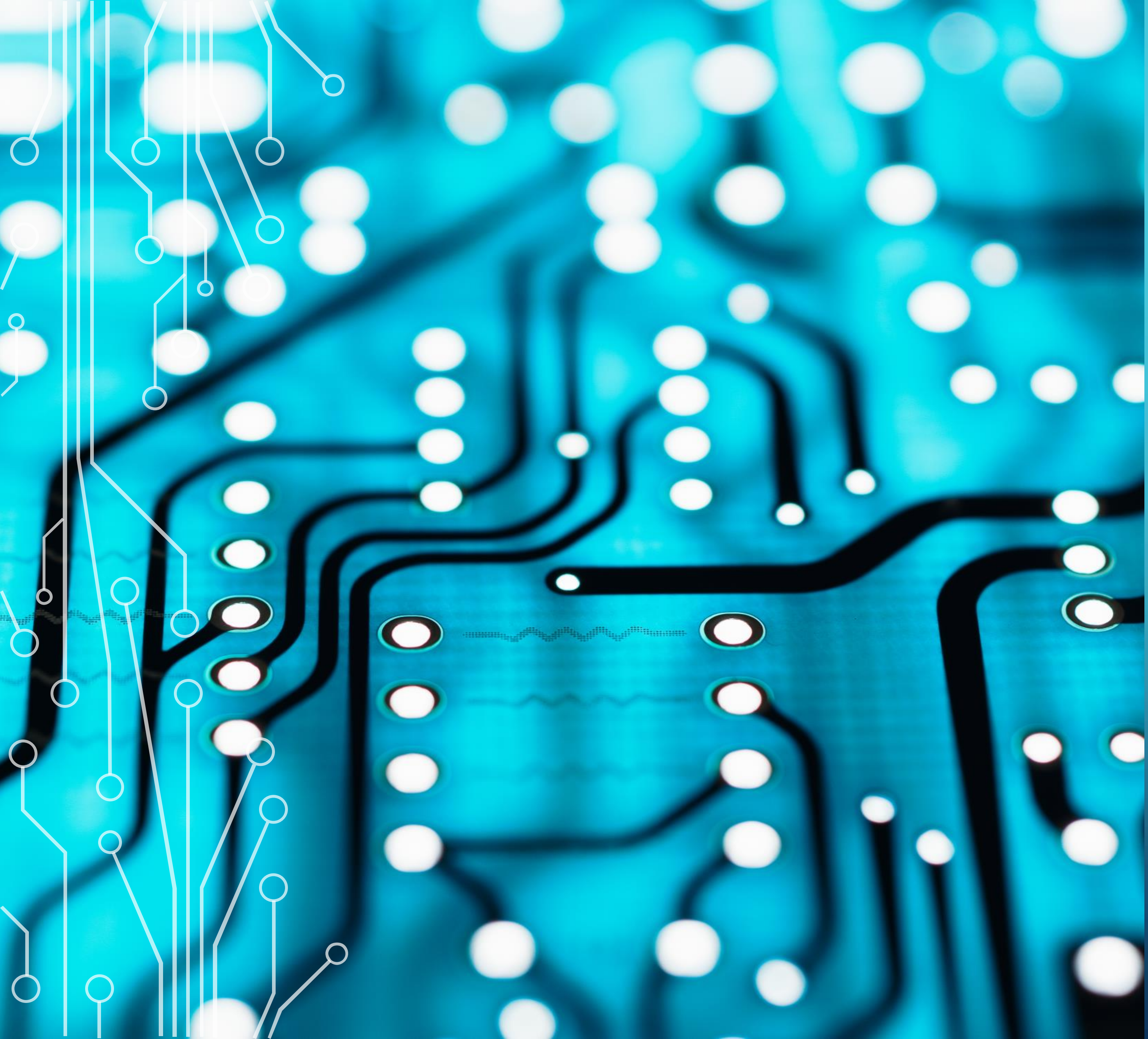
PREDICTING CUSTOMER CHURN IN THE CREDIT CARD INDUSTRY

BY JAMES GARCIA, DAVID WILLIAMS,

SVETLANA KACHINA & ILKNUR TOPTAS



PROJECT OVERVIEW

- Problem: Churn costs banks millions annually
- Goal: Predict customers at risk of account cancellation
- Dataset: 10,000+ customers
- Solution: Machine learning classification models
- Value: Early intervention = improved retention, reduced losses

STAKEHOLDERS & THEIR NEEDS

Marketing

Personalized retention campaigns

Customer Success Managers

Early outreach to high-risk clients

Product Managers

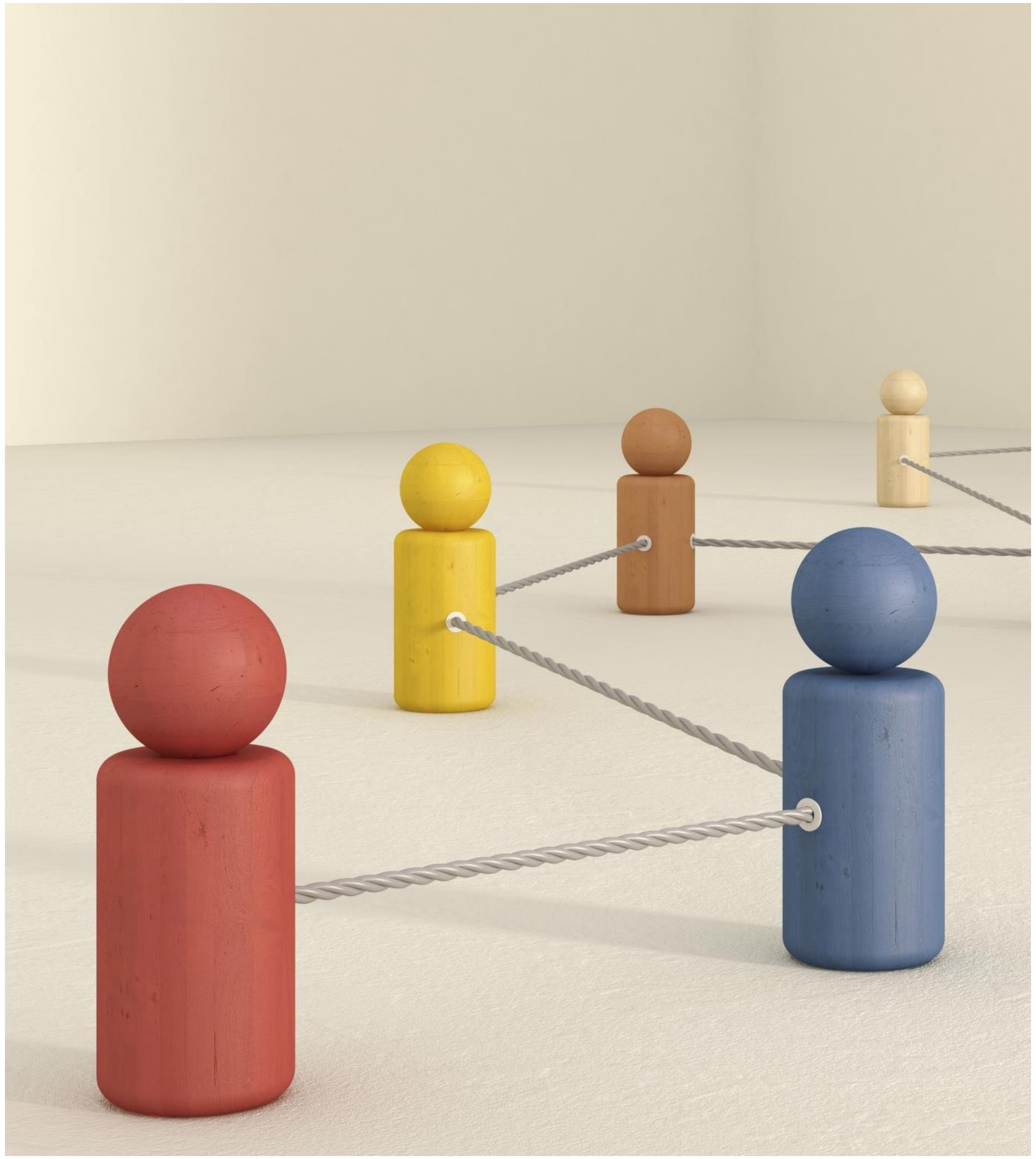
Identify attrition-linked product gaps

Data Analysts

Continuous churn analysis & reporting

Executive Leadership

Strategy decisions, churn monitoring



KEY QUESTIONS WE ADDRESS

- Who is most likely to churn?
- Why are they leaving?
- How can we intervene early?
- Can we group customers by churn risk?
- Is our prediction model reliable?

DATASET & FEATURES

- Source: Gigasheet – 10,000+ credit card customers
- Target variable: Churn (derived from Attrition_Flag)
- Key features:
  - Demographics: Gender, Income
  - Account info: Credit Limit, Revolving Balance
  - Behavior: Total Transaction Count, Amount, and Change

	CLIENTNUM	Attrition_Flag	Customer_Age	Gender	Dependent_count	Education_Level	Marital_Status	Income_Category	Card_Categ
0	768805383	Existing Customer	45	M	3	High School	Married	$60K$—80K	B
1	818770008	Existing Customer	49	F	5	Graduate	Single	Less than $40K	B
2	713982108	Existing Customer	51	M	3	Graduate	Married	$80K$—120K	B
3	769911858	Existing Customer	40	F	4	High School	Unknown	Less than $40K	B
4	709106358	Existing Customer	40	M	3	Uneducated	Married	$60K$—80K	B

TECHNOLOGIES USED

- Python – Modeling & analysis
- Pandas, NumPy – Data wrangling
- Seaborn, Matplotlib – Visualization
- Scikit-learn – Machine Learning
- AWS RDS (PostgreSQL) – Cloud database
- GitHub – Collaboration

DATA CLEANING

```
raw_df.drop(columns=[
    'Naive_Bayes_Classifier_Attrition_Flag_Card_Category_Contacts_Count_12_mon_Dependent_count_Education_Level_Months_Inact:
    'Naive_Bayes_Classifier_Attrition_Flag_Card_Category_Contacts_Count_12_mon_Dependent_count_Education_Level_Months_Inact:
], inplace=True)

edu_map = {'Uneducated': 0, 'High School': 1, 'College': 2, 'Graduate': 3, 'Post-Graduate': 4, 'Doctorate': 5, 'Unknown': -1
marital_map = {'Single': 0, 'Married': 1, 'Divorced': 2, 'Unknown': -1}
income_map = {'Less than $40K': 0, '$40K - $60K': 1, '$60K - $80K': 2, '$80K - $120K': 3, '$120K +': 4, 'Unknown': -1}
card_map = {'Blue': 0, 'Silver': 1, 'Gold': 2, 'Platinum': 3}
gender_map = {'M': 1, 'F': 0}

raw_df['Education_Level'] = raw_df['Education_Level'].map(edu_map)
raw_df['Marital_Status'] = raw_df['Marital_Status'].map(marital_map)
raw_df['Income_Category'] = raw_df['Income_Category'].map(income_map)
raw_df['Card_Category'] = raw_df['Card_Category'].map(card_map)
raw_df['Gender'] = raw_df['Gender'].map(gender_map)
raw_df['Churn'] = raw_df['Attrition_Flag'].map({'Existing Customer': 1, 'Attrited Customer': 0})
raw_df.drop(columns=['Attrition_Flag'], inplace=True)
df = raw_df.copy()
```

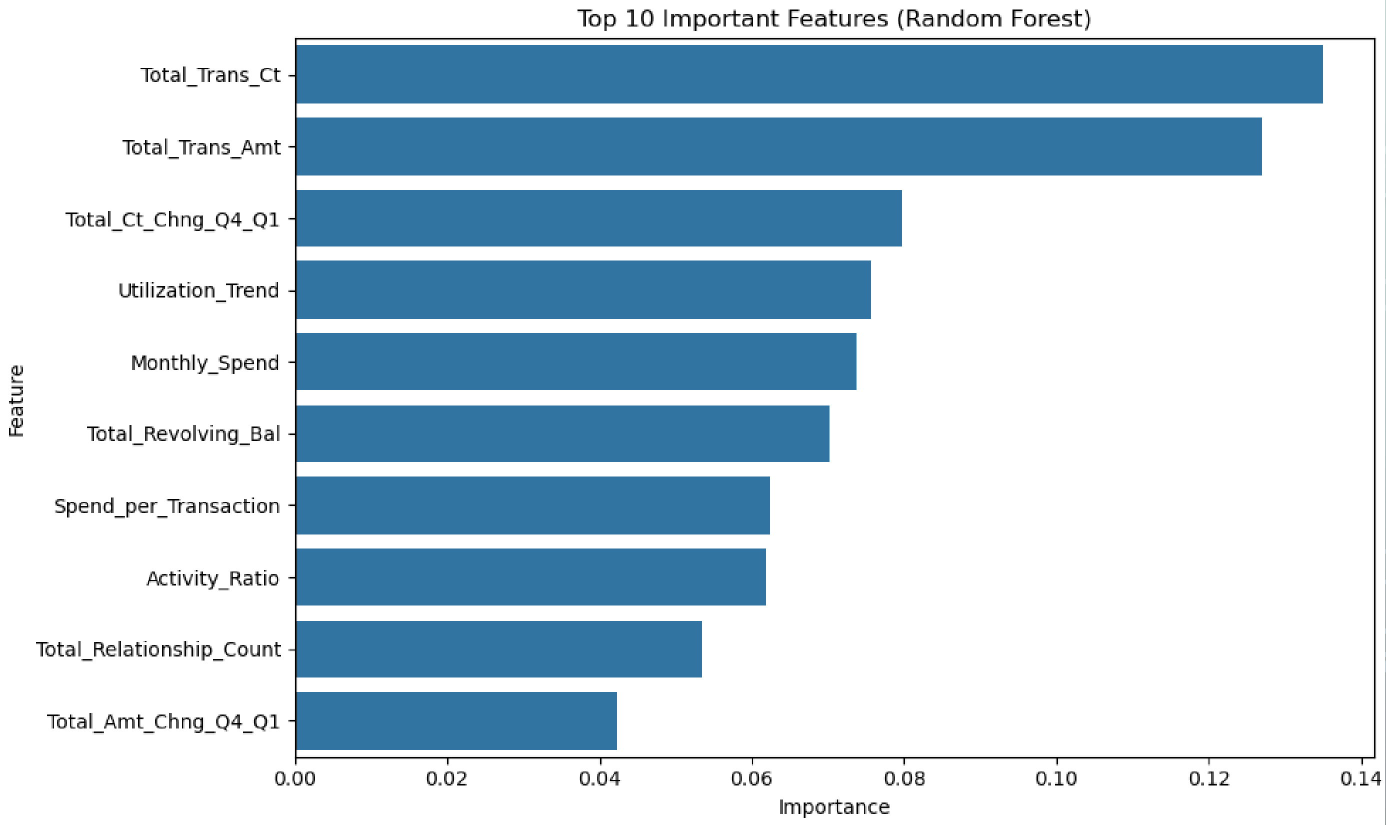
	CLIENTNUM	Customer_Age	Gender	Dependent_count	Education_Level	Marital_Status	Income_Category	Card_Category	Months_on
0	768805383	45	1	3	1	1	2	0	
1	818770008	49	0	5	3	0	0	0	
2	713982108	51	1	3	3	1	3	0	
3	769911858	40	0	4	1	-1	0	0	
4	709106358	40	1	3	0	1	2	0	

MODELING APPROACH

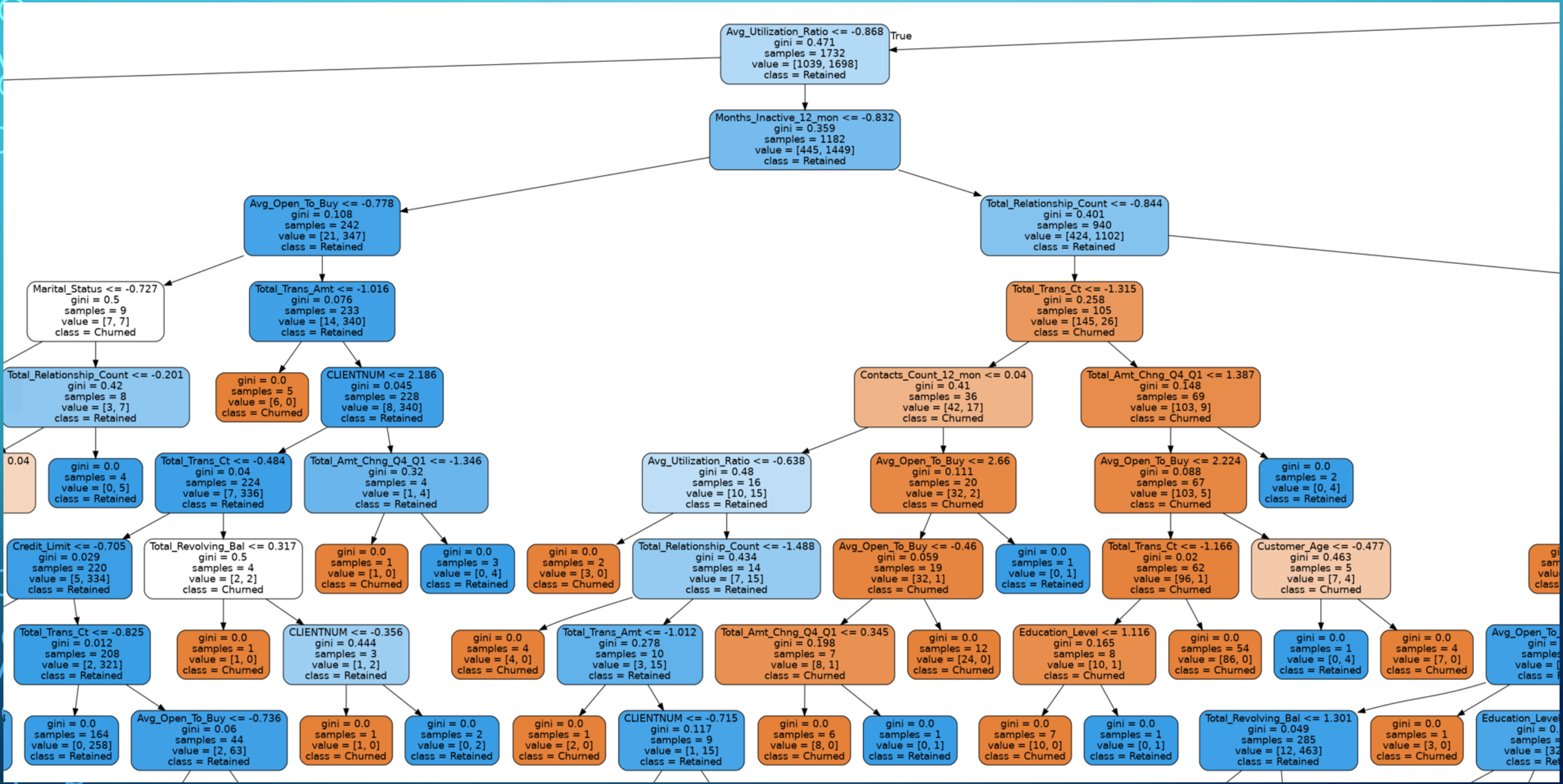
Step 2: Feature Engineering

We create new behavioral features to improve model accuracy.

```
df['Spend_per_Transaction'] = df['Total_Trans_Amt'] / df['Total_Trans_Ct']
df['Monthly_Spend'] = df['Total_Trans_Amt'] / df['Months_on_book']
df['Utilization_Trend'] = df['Avg_Utilization_Ratio'] * df['Credit_Limit']
df['Activity_Ratio'] = df['Total_Ct_Chng_Q4_Q1'] * df['Total_Amt_Chng_Q4_Q1']
```



MODEL DEVELOPMENT



MODEL RESULTS

TASK: BINARY CLASSIFICATION (CHURN / NO CHURN)

MODELS TESTED:
LOGISTIC REGRESSION

DECISION TREE
✅ RANDOM FOREST (BEST PERFORMER)

HYPERPARAMETER TUNING: GRIDSEARCHCV

EVALUATION: ACCURACY, RECALL, F1 SCORE

Model	Accuracy	Recall	F1 Score
✅ Random Forest	96.10%	98.30%	97.69%
Decision Tree	94.13%	96.06%	96.49%
Logistic Regression	91.31%	96.94%	94.93%

Top predictors:
Total_Trans_Ct,
Total_Trans_Amt,
Total_Ct_Chng_Q4_Q1

KEY INSIGHTS & BUSINESS VALUE

Customer behavior — especially transaction count, balance usage, and tenure — predicts churn better than demographics.

High churn-risk customers show warning signs early, offering a critical window for retention.

RETENTION: EARLY IDENTIFICATION ALLOWS OUTREACH BEFORE CHURN OCCURS, BOOSTING CUSTOMER LIFETIME VALUE (CLV).

TARGETING: ENABLES MARKETING TO FOCUS EFFORTS ON USERS MOST LIKELY TO LEAVE.

STRATEGIC DECISIONS: EMPOWERS EXECUTIVES WITH DATA-BACKED CHURN KPIS.

SCALABILITY: CLOUD-BASED MODEL STRUCTURE (AWS RDS) ENABLES EASY RETRAINING AND DEPLOYMENT.

CONCLUSION

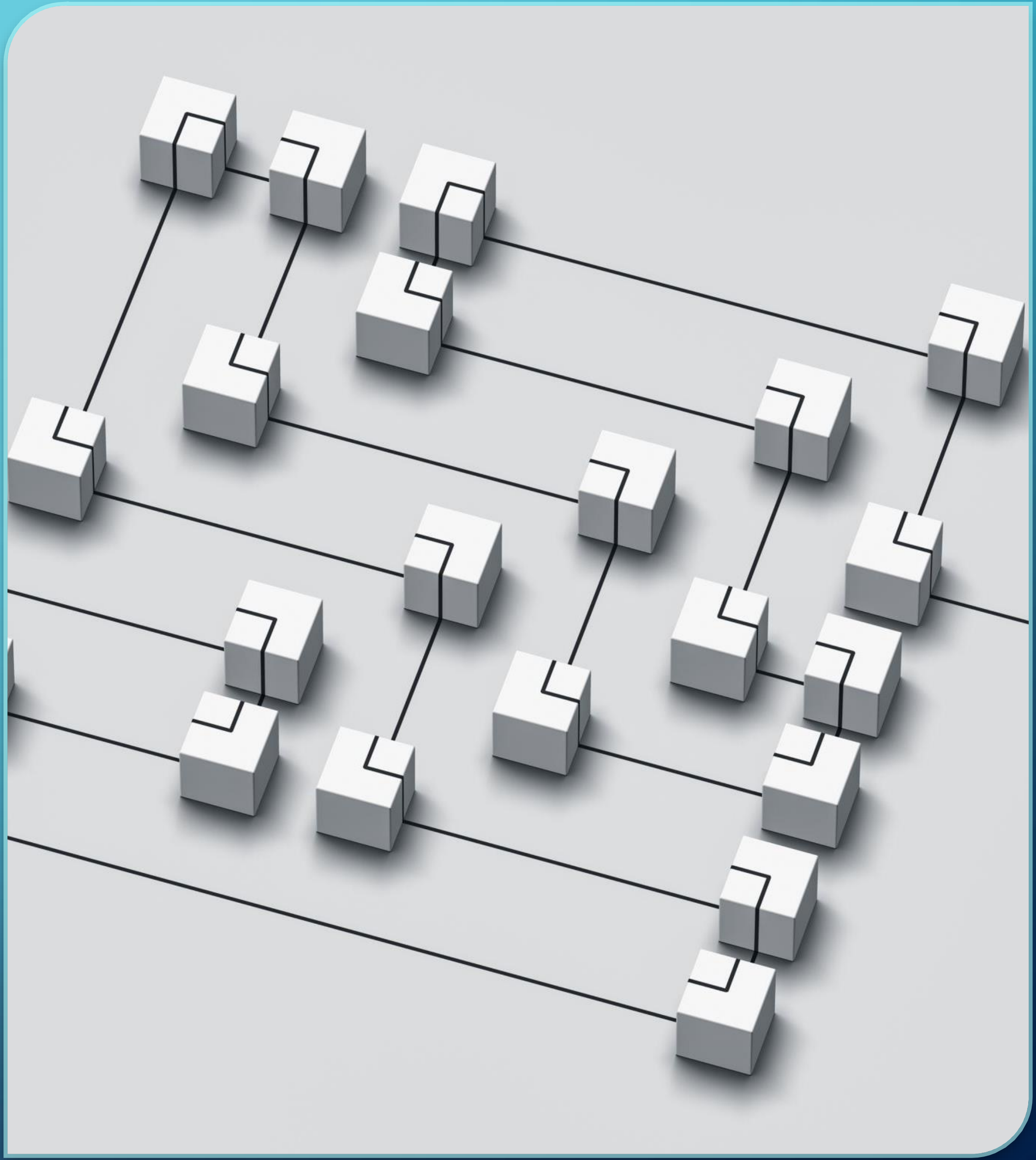
What We Achieved:

- Built a machine learning model achieving 96.1% accuracy and 98.3% recall.
- Identified top predictors of churn: transaction volume, credit usage, and customer tenure.
- Developed a cloud-integrated, scalable, and reproducible pipeline.

Predicting churn is not just preventing loss — it's building stronger, smarter customer relationships.

NEXT STEPS FOR THE PROJECT

- Develop a real-time dashboard or interactive app where predictions can be visualized live
- Compare our current models with advanced techniques such as neural networks or ensemble stacking to see if accuracy improves
- Integrating this prediction pipeline with a CRM system could trigger automated retention emails or alerts
- Continue adding newer customer data to keep the model fresh and responsive to behavior trends
- Turn this into a fully deployable customer retention tool



Q&A