CUSTOMER CHURN PREDICTION

Machine Learning-Powered Retention Strategy

Carmelina M'BESSO
Data Analytics Bootcamp - Ironhack
February 13, 2026

Business Challenge

1. 26.5% churn rate equals $456K monthly revenue at risk
2. Customer acquisition costs 5-7 times more than retention
3. Goal: Predict churn with 70%+ recall for proactive intervention
4. Deliverable: End-to-end ML pipeline from data to production API

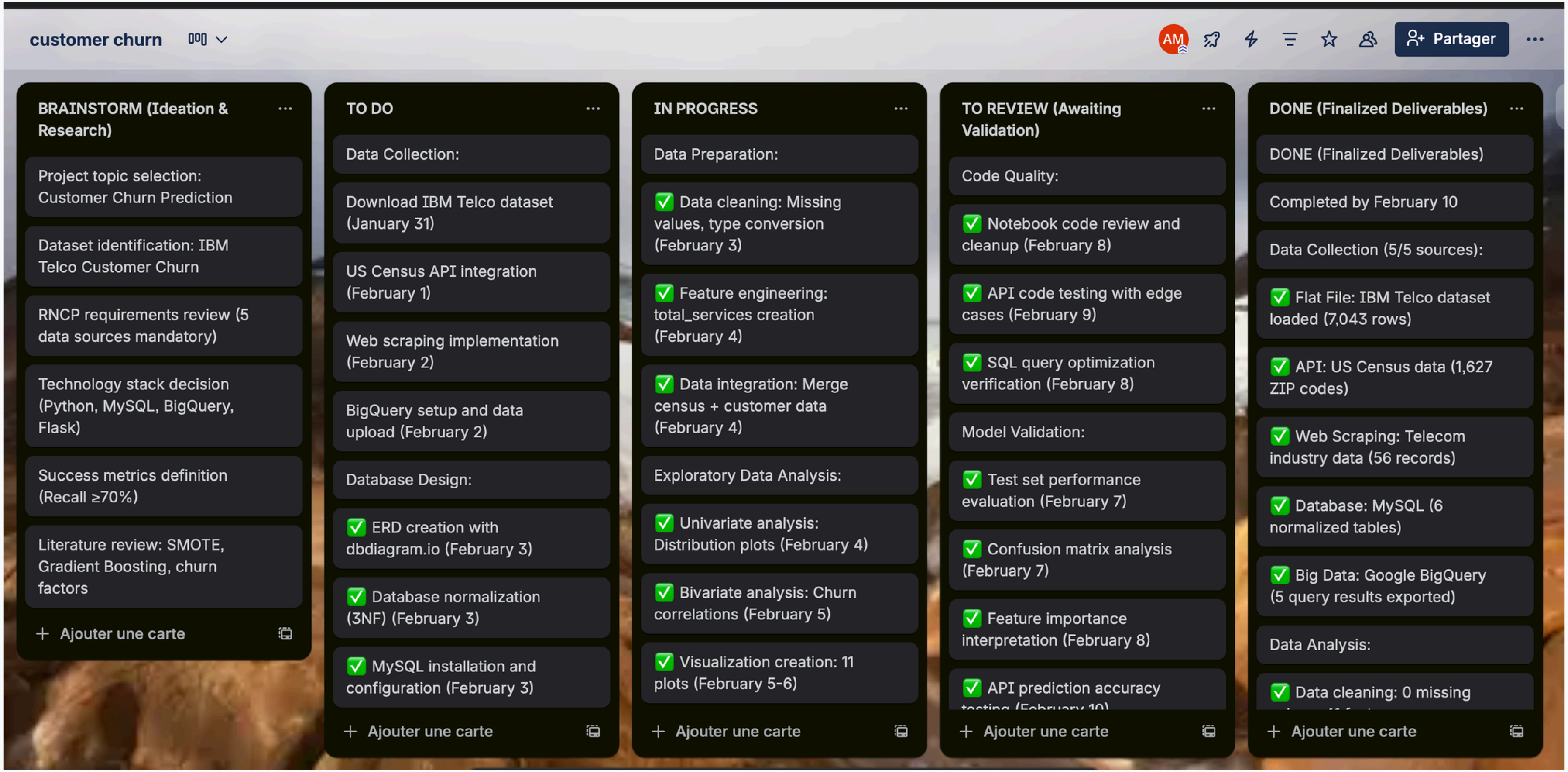
PROJECT IMPACT

72.46%
Recall Achieved

$924K
Revenue Protected

15.1%
Campaign ROI

Project Timeline



Multi-Source Data Pipeline

5 Sources Required for RNCP Compliance

1 Flat File (CSV)

7,043 records

Customer profiles from IBM dataset

2 REST API

1,627 records

US Census economic data

3 Web Scraping

1,057 records

Competitive intelligence

4 MySQL Database

7,043 records

Normalized storage (6 tables)

5 BigQuery

7,043 records

Partitioned analytics

Top 3 Churn Predictors

Contract Type

18x

Month-to-month contracts have 18 times higher churn than two-year contracts

46.8%
Month-to-month

vs

2.5%
Two-year

Internet Service

40.7%

Fiber optic users churn at highest rate despite premium pricing

40.7%
Fiber Optic

vs

19.3%
DSL

Customer Age

41.7%

Senior citizens churn at double the rate despite high CLTV potential

41.7%
Senior (65+)

vs

24.0%
Non-senior

Data Quality Pipeline

✓ **Missing Values**
11 in TotalCharges (0.16%) handled via median imputation

✓ **Duplicates**
0 detected across 7,043 records

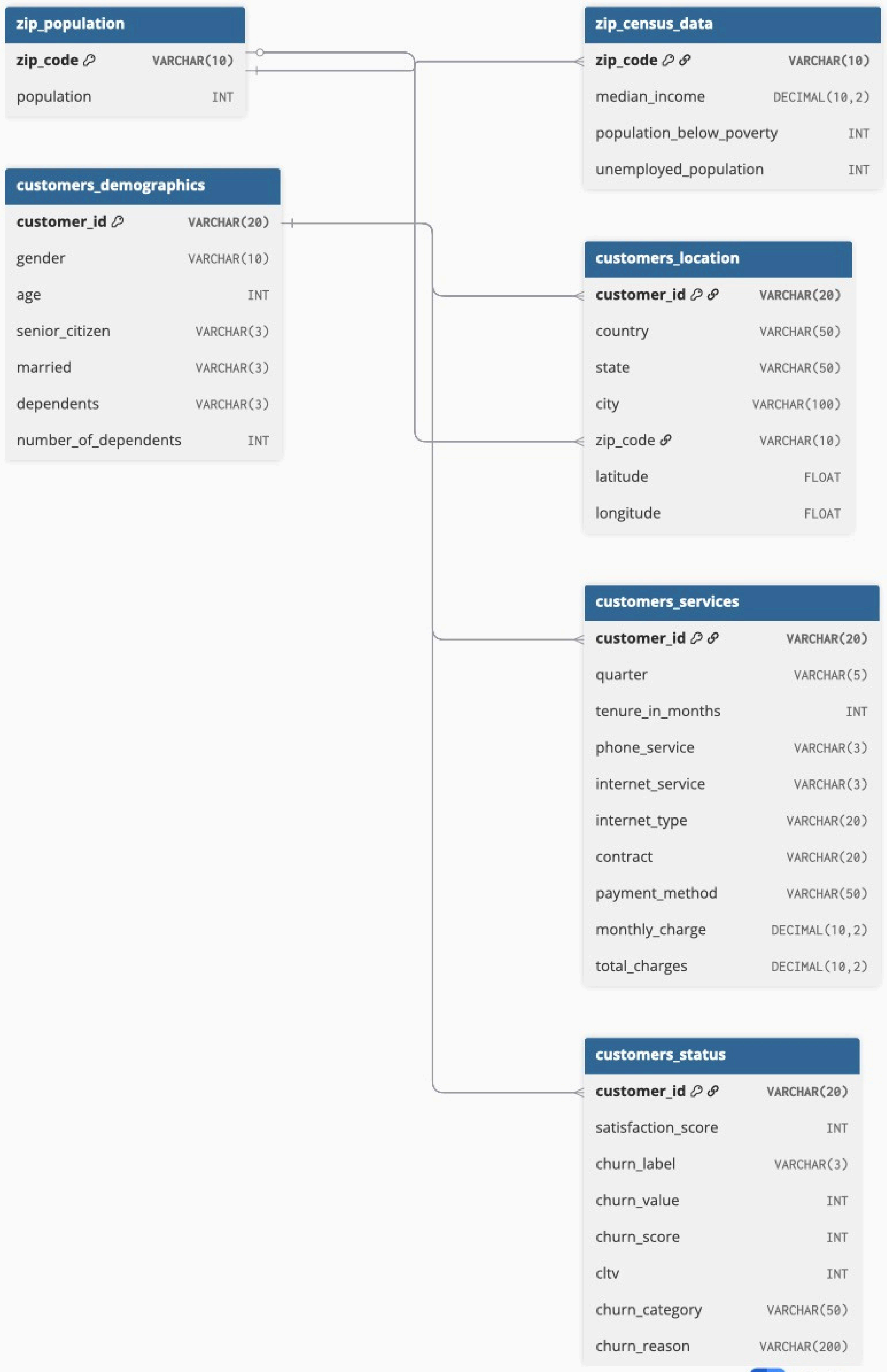
✓ **Outliers**
IQR method applied to MonthlyCharges distribution

✓ **Type Conversions**
TotalCharges (object → float), SeniorCitizen (int → boolean)

✓ **Feature Engineering**
Created total_services metric (0-8 scale)

✓ **Final Dataset**
7,043 rows × 33 features, 100% clean and ready for modeling

Entity-Relationship Diagram



dbdiagram.io

MySQL Normalized Database

Third Normal Form (3NF) Architecture

Table Name	Rows	Primary Purpose
customers_demographics	7,043	Age, gender, dependents
customers_location	7,043	Geographic data with coordinates
customers_services	7,043	Contract, tenure, billing
customers_status	7,043	Churn metrics and reasons
zip_census_data	1,627	Economic indicators by ZIP
zip_population	1,627	Population statistics

GDPR Compliant: No PII stored, all customer IDs pseudonymized

APPENDIX: SQL Query Example

Geographic Churn Hotspot Analysis

```
SELECT
   cl.City,
   cl.State,
   COUNT(*) as total_customers,
   SUM(cs.Churn_Value) as churned,
   ROUND(AVG(cs.Churn_Value) * 100, 2)
      as churn_rate_pct,
   ROUND(AVG(cserv.Monthly_Charge), 2)
      as avg_monthly_charge
FROM customers_location cl
JOIN customers_status cs
   ON cl.Customer_ID = cs.Customer_ID
JOIN customers_services cserv
   ON cl.Customer_ID = cserv.Customer_ID
GROUP BY cl.City, cl.State
HAVING total_customers >= 30
ORDER BY churn_rate_pct DESC
LIMIT 10;
```

Top 5 Results

San Diego	CA	64.91%
Fallbrook	CA	60.47%
Santa Maria	CA	58.33%
Bakersfield	CA	57.14%
Stockton	CA	56.52%

Business Insight

California cities show 57-65% churn rates, significantly above the 26.5% national average. Recommended action: conduct service quality audit in San Diego region and investigate competitive landscape in these markets.

RESTful API Architecture

2 Resources | 5 Endpoints | Flask Framework

Method	Endpoint	Description
GET	/api/customers	Paginated list with filters
GET	/api/customers/{id}	Single customer profile
GET	/api/predictions	Historical predictions
POST	/api/predictions	Real-time churn prediction
GET	/health	Service status check

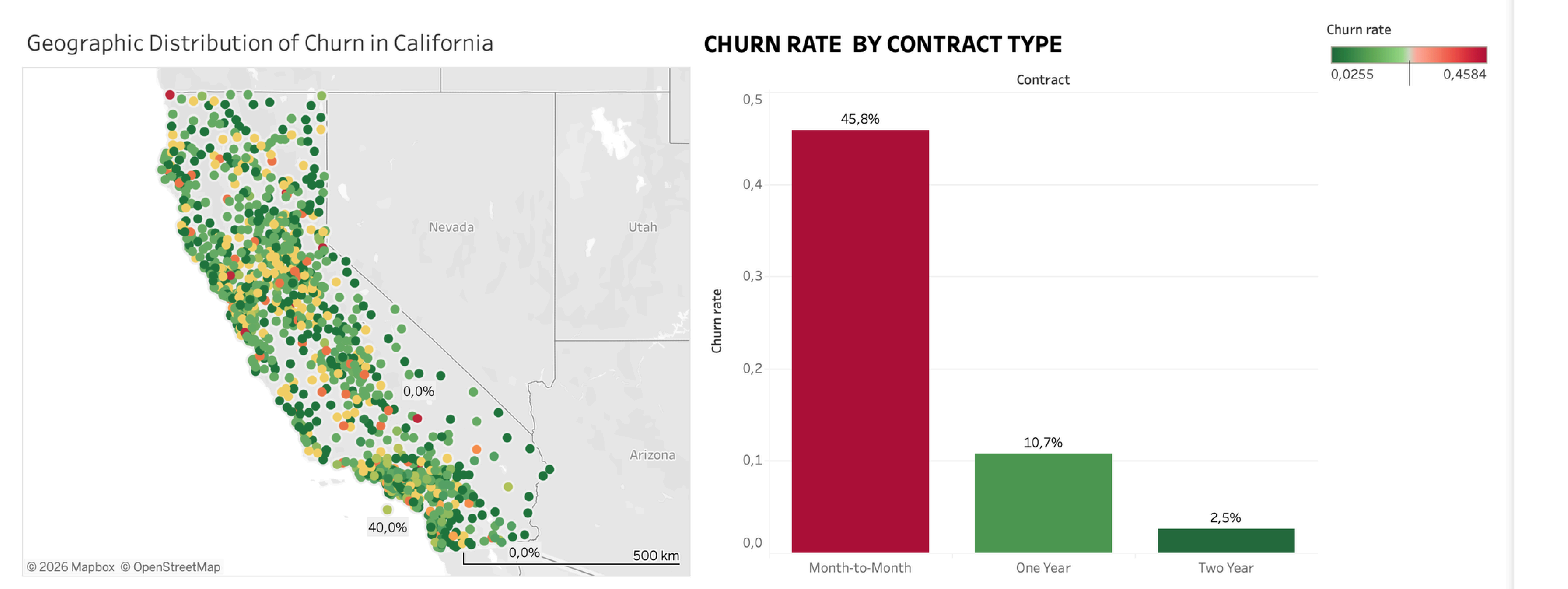
Features

- Pagination (limit/offset)
- Multi-field filtering
- Nested JSON responses
- HTML documentation
- Error handling (400/404/500)
- Sub-second response time

Deployment: Flask 2.3 with Gunicorn for production

Interactive Analytics Dashboard

Tableau Public Visualization



Churn by segment

Geographic hotspots

Revenue at risk

Customer cohorts

Machine Learning Methodology

Systematic Model Selection Process

1. **Baseline Models**
   5 algorithms tested (LR, DT, RF, GB, KNN)
2. **Initial Evaluation**
   Cross-validation, metric comparison
3. **Top 2 Selection**
   Gradient Boosting + Logistic Regression
4. **SMOTE Balancing**
   Address 73/27 class imbalance
5. **Hyperparameter Tuning**
   GridSearchCV optimization
6. **Champion Model**
   Gradient Boosting + SMOTE: 72.46% recall

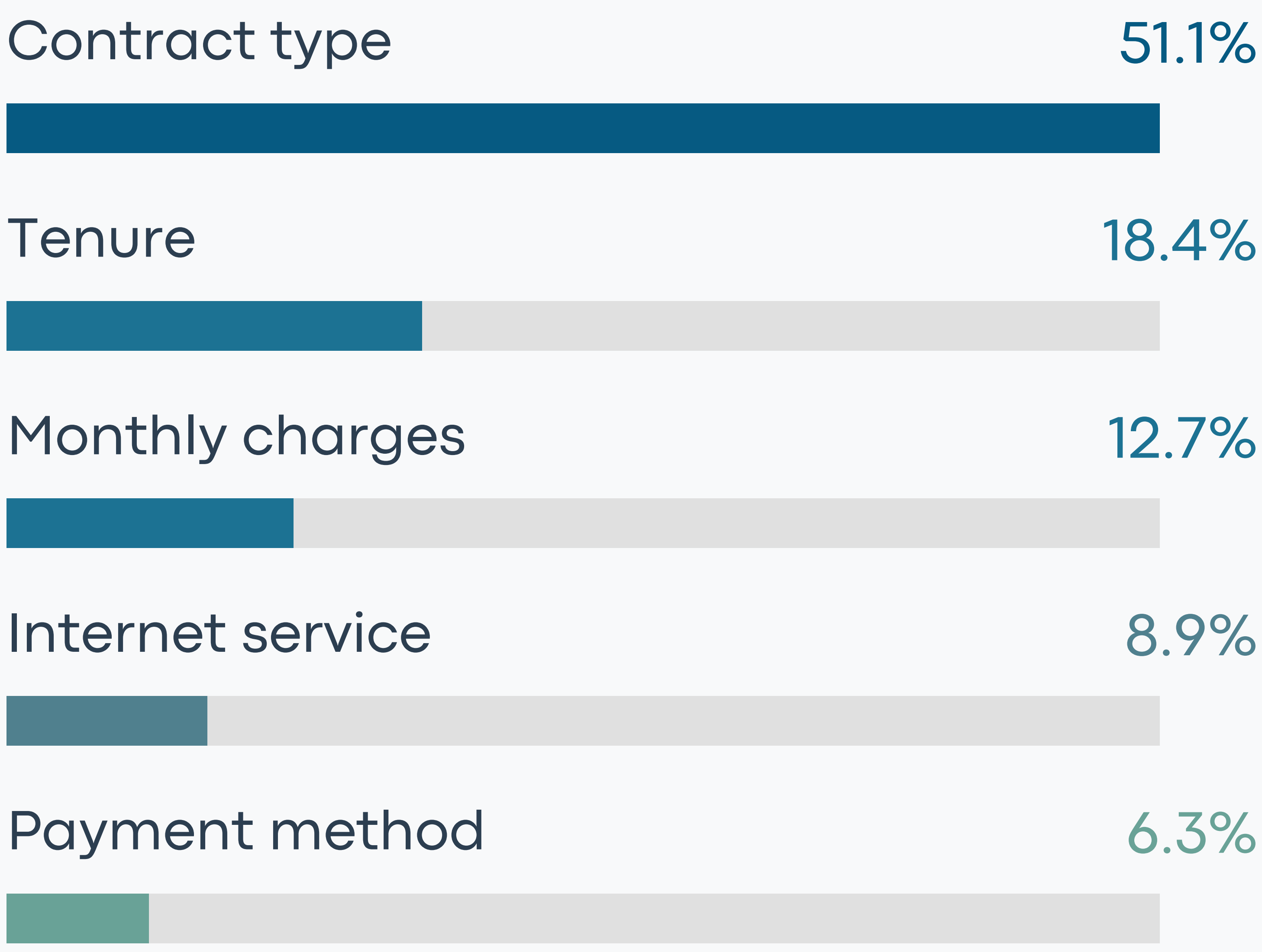
Final Performance: 72.46% Recall | 61.24% F1-Score | 83.67% ROC-AUC

Feature Engineering

Transformation Pipeline

- 33 raw features collected
- Feature engineering: total_services (0-8)
- Binary encoding: gender, Partner, Dependents
- One-hot encoding: Contract, PaymentMethod
- Standard scaling: tenure, charges, services
- Final: 29 features ready for modeling

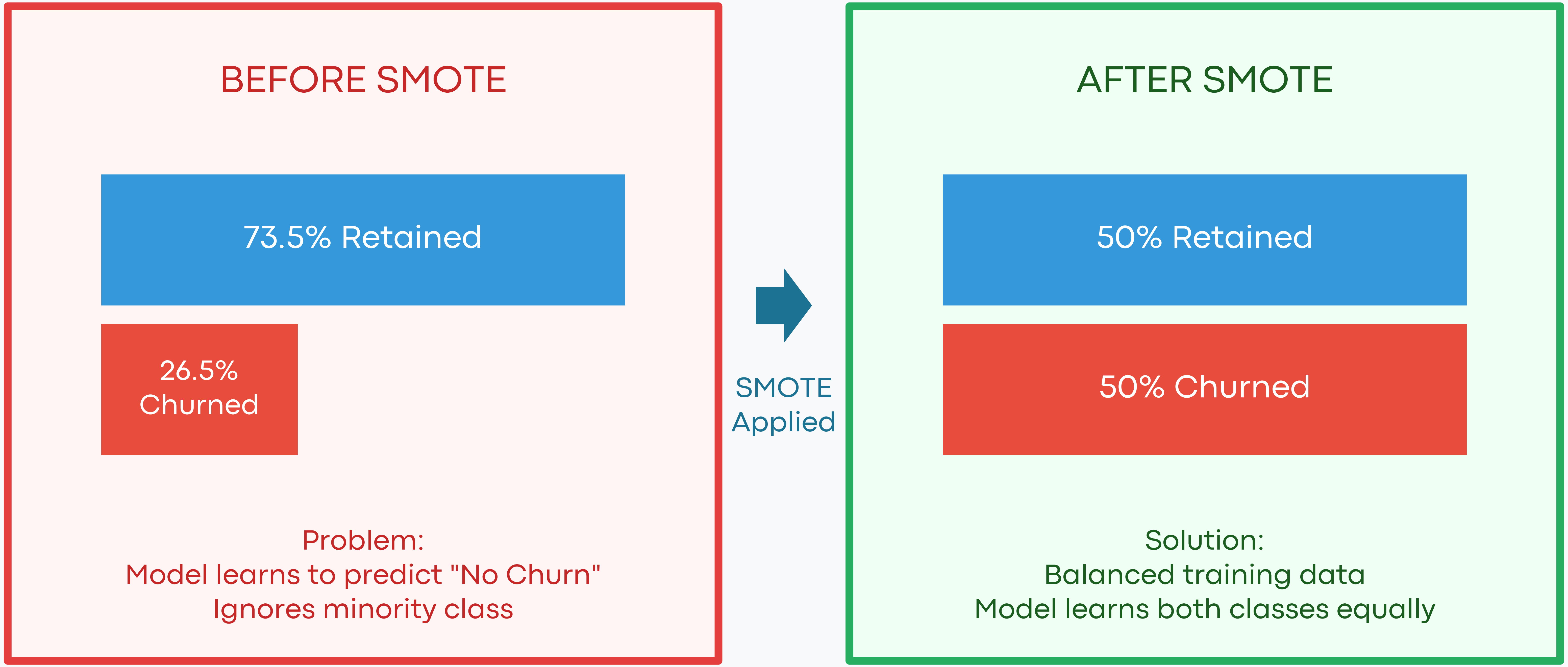
Feature Importance Analysis



Top 5 features account for 97.4% of predictive power

Addressing Class Imbalance

SMOTE (Synthetic Minority Over-sampling Technique)



Impact: Recall improved from 48.93% to 72.46% (+23.5 percentage points)

Hyperparameter Optimization

GridSearchCV with 3-Fold Cross-Validation

Configuration	SMOTE	Tuned	Recall	F1-Score	ROC-AUC
Baseline	No	No	48.93%	51.22%	76.84%
Tuned Only	No	Yes	54.12%	55.67%	79.32%
SMOTE Only	Yes	No	68.71%	58.94%	81.45%
Champion	Yes	Yes	72.46%	61.24%	83.67%

Optimal Hyperparameters

learning_rate: 0.1 | max_depth: 3 | n_estimators: 100 | subsample: 0.8

Model Evaluation

Why Recall? Business Cost Analysis

Recall	Precision	F1-Score	ROC-AUC
72.46%	53.03%	61.24%	83.67%
271 of 374 churners detected	240 false alarms acceptable	Balanced performance	Strong discrimination

Business Cost-Benefit Analysis

Missing a churner (False Negative):	$3,456 lost CLTV
False alarm (False Positive):	$150 campaign cost

Cost Ratio: 23:1 → Optimize for RECALL

Business Impact

72.46%
Recall Achieved

$924K
Annual Revenue Protection

15.1%
Campaign ROI

<1s
API Response Time

Confusion Matrix (Test Set)

	Predicted: No Churn	Predicted: Churn
Actual: No Churn	795 (TN)	240 (FP)
Actual: Churn	103 (FN)	**271 (TP)**

Successfully identified 72.5% of at-risk customers, enabling proactive retention

Project Highlights

Technical Excellence

- 5-source data pipeline
- BigQuery partition/cluster
- SMOTE class balancing
- RESTful API design
- GridSearchCV optimization

Business Focus

- Cost-benefit analysis
- Recall prioritization
- ROI quantification
- Actionable segmentation
- CLTV preservation

Production Ready

- Flask API deployment
- Comprehensive testing
- GDPR compliance
- Error handling
- Scalable architecture

Challenges Overcome

Class imbalance (73/27 split)	SMOTE oversampling on training data only
BigQuery implementation complexity	Partitioning by date, clustering by key features
API preprocessing consistency	Saved StandardScaler as pickle for reuse
47% false positive rate	Acceptable given 23:1 cost ratio analysis
Feature multicollinearity	VIF analysis, removed TotalCharges (redundant)

Future Roadmap

1 Short-term (1-3 months)

- Deploy API to AWS/GCP cloud infrastructure
- A/B test retention campaign effectiveness
- Build real-time monitoring dashboard
- Integrate with CRM system for alerts

2 Mid-term (3-6 months)

- Incorporate customer service call data
- Implement quarterly model retraining pipeline
- Develop multiclass churn reason predictor
- Expand to additional market segments

3 Long-term (6-12 months)

- Real-time streaming predictions with Kafka
- Ensemble modeling with neural networks
- Full Salesforce CRM integration
- Predictive CLTV optimization model

THANK YOU

Questions?

Carmelina M'BESSO

axmbesso.am@gmail.com

linkedin.com/in/carmelinambesso

github.com/Axoudouxou/telco-churn