Clustering FinTech Users: From Data to Empathy

Advanced Clustering Techniques on Simulated Financial Data
10,000 Users, 12 Features, 7 Natural Segments
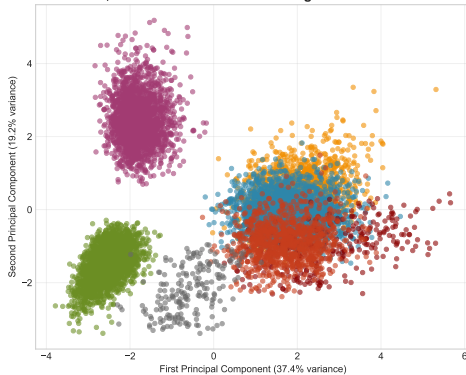
Week 2: Machine Learning for Smarter Innovation

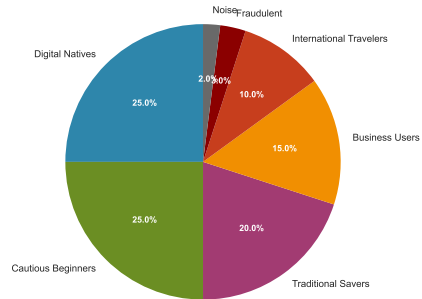BSc Course - MSc-Level Dataset Analysis

2025

Note: Using SIMULATED data for educational purposes

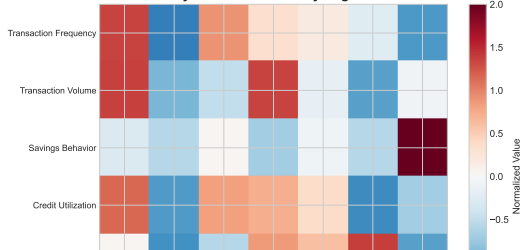# FinTech Dataset Overview: 10,000 Users, 12 Features, 7 Segments

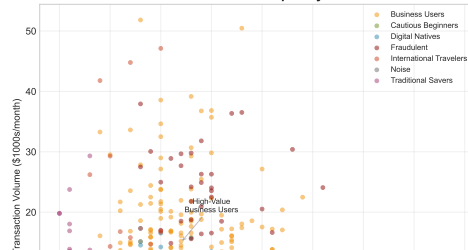**10,000 FinTech Users: Natural Segments Revealed**

Legend:
- Business Users
- Cautious Beginners
- Digital Natives
- Fraudulent
- International Travelers
- Noise
- Traditional Savers

First Principal Component (37.4% variance)
Second Principal Component (19.2% variance)

User Segment Distribution

- Digital Natives 25.0%
- Cautious Beginners 25.0%
- Traditional Savers 20.0%
- Business Users 15.0%
- International Travelers 10.0%
- Fraudulent 3.0%
- Noise 2.0%

Key Feature Patterns by Segment

- Transaction Frequency
- Transaction Volume
- Savings Behavior
- Credit Utilization

Normalized Value (scale −0.5 to 2.0)

Behavioral Patterns: Frequency vs Volume

Legend:
- Business Users
- Cautious Beginners
- Digital Natives
- Fraudulent
- International Travelers
- Noise
- Traditional Savers

High-Value Business Users

Transaction Volume ($1000s/month)

The Scenario

- 10,000 simulated FinTech users
- Complex behavioral patterns
- Hidden segments to discover
- Fraud patterns embedded
- Realistic business challenges

Learning Objectives

- Apply 4 clustering algorithms
- Validate cluster quality
- Detect anomalies
- Create personas

Why This Dataset?

- Industry-relevant features
- Multiple clustering challenges
- Real-world complexity
- MSc-level technical depth
- Business value demonstration

Simulated data with real-world patterns

Transaction Metrics
- Frequency (0-39/day)
- Volume ($0.75-90K)
- Peak hours (0-100%)
- Categories (1-28 types)

Financial Behavior
- Savings (0-280 score)
- Credit use (0-143%)
- International (0-100%)
- Payment types (1-21)

Engagement Patterns
- Session time (0-84 min)
- Support (0-10 contacts)
- Devices (0-17 switches)
- Age (0-2895 days)

All features synthetically generated with realistic distributions

Technical Skills

- Handling skewed distributions
- Missing data (0.46% NaN)
- Feature scaling strategies
- Distance metric selection
- Validation techniques
- Scalability considerations

Industry Context

- Similar to PayPal, Revolut data
- KYC/AML requirements
- Personalization at scale
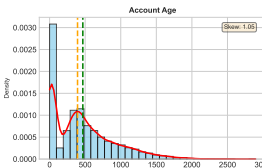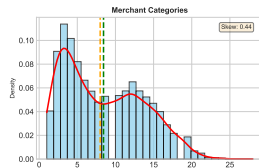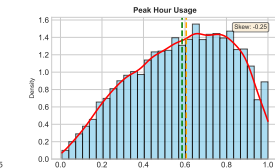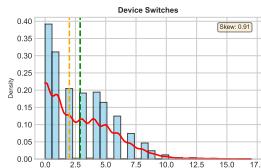- Fraud detection needs

Business Applications

- Customer segmentation
- Risk assessment
- Product recommendations
- Churn prediction
- Support optimization
- Marketing targeting

Career Preparation

- Data Scientist roles
- ML Engineer positions
- Business Analyst tracks
- FinTech opportunities
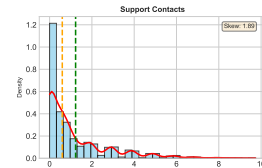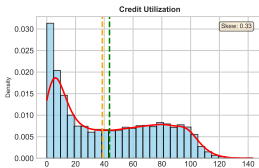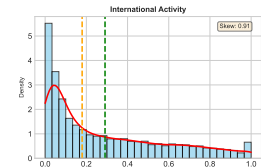
FinTech Dataset Quality Report (Simulated Data)

Missing Data Analysis

international_activity — 1.74%
device_switches — 1.89%
credit_utilization — 1.95%

(Missing %)

Summary Statistics Table

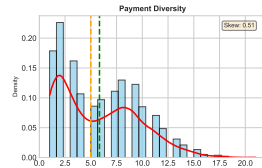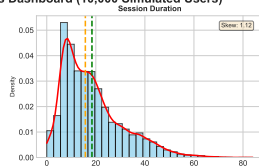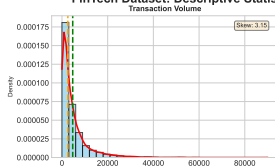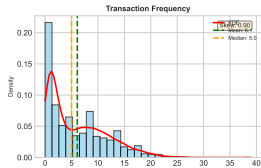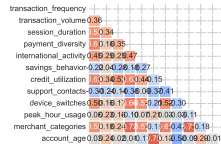|	Mean	Std Dev	Min	Max	CV
Transaction Frequenc	6.09	5.5	0.0	39.0	0.9
Transaction Volume	4705.02	6206.91	0.75	90136.96	1.32
Session Duration	18.42	12.26	0.0	83.99	0.67
Payment Diversity	5.84	3.82	1.0	21.0	0.65
International Activi	0.29	0.28	0.0	1.0	0.98
Savings Behavior	36.85	44.66	0.0	279.56	1.21
Credit Utilization	43.61	34.57	0.0	143.21	0.79
Support Contacts	1.23	1.61	0.0	9.74	1.31
Device Switches	2.89	2.62	0.0	17.0	0.91
Peak Hour Usage	0.59	0.24	0.01	1.0	0.4
Merchant Categories	8.44	5.2	1.0	28.0	0.62
Account Age	467.08	430.56	0.0	2894.87	0.92
True Label	2.07	1.7	0.0	6.0	0.82

Segment Distribution

Pie chart segments:
- Noise 2.0%
- Fraudulent 3.0%
- International Travelers 10.0%
- Digital Natives 25.0%
- Business Users 15.0%
- Traditional Savers 20.0%
- Cautious Beginners 25.0%
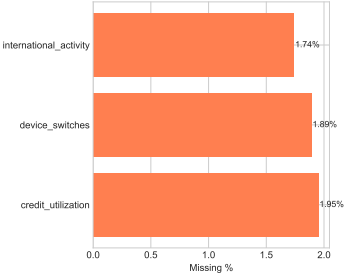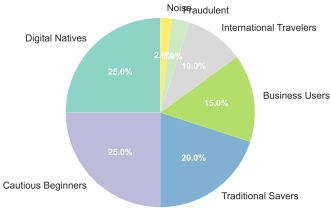
Feature Variability (Top 10)

- Transaction Volume
- Support Contacts
- Savings Behavior
- International Activi
- Account Age
- Device Switches
- Transaction Frequenc
- True Label
- Credit Utilization
- Session Duration

(Coefficient of Variation)
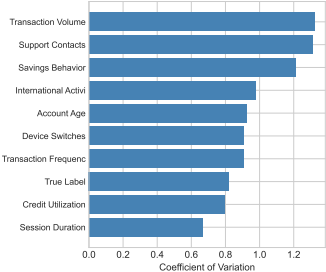
```
DATASET INFORMATION
====================

Total Samples: 10,000
Features: 12
Segments: 7

Data Type: SIMULATED
Purpose: Educational

Segment Breakdown:
------------------

Digital Natives: 2,500 (25.0%)
Cautious Beginners: 2,500 (25.0%)
Traditional Savers: 2,000 (20.0%)
Business Users: 1,500 (15.0%)
International Travelers: 1,000 (10.0%)
Fraudulent: 300 (3.0%)
Noise: 200 (2.0%)

Missing Data: 558 values
(0.40% of total)

Note: This dataset was synthetically
generated to demonstrate clustering
techniques for FinTech applications.
```

Part 2: Advanced Clustering Techniques

Comparing 4 Algorithms on FinTech Data

Clustering Algorithm Comparison on FinTech Dataset

Ground Truth Segments

Legend:
- Business Users
- Cautious Beginners
- Digital Natives
- Fraudulent
- International Travelers
- Noise
- Traditional Savers

K-Means (k=5)
Silhouette: 0.348

- Cluster 0
- Cluster 1
- Cluster 2
- Cluster 3
- Cluster 4

DBSCAN
Silhouette: 0.438

Clusters: 7
Noise: 7245 (72.5%)
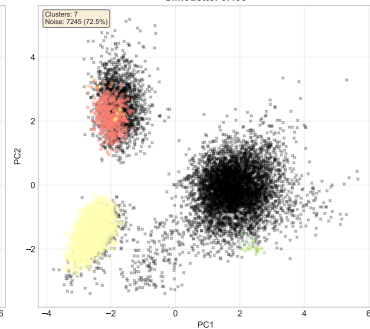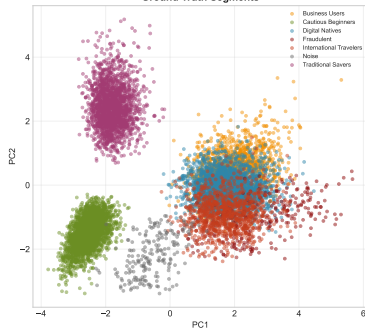
Hierarchical
Silhouette: 0.378

- Cluster 0
- Cluster 1
- Cluster 2
- Cluster 3
- Cluster 4
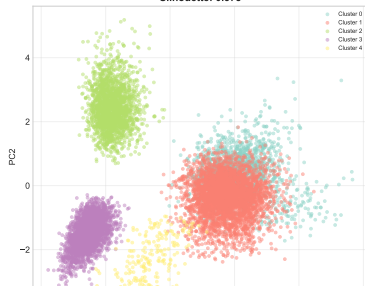
GMM (5 components)
Silhouette: 0.345

- Cluster 0
- Cluster 1
- Cluster 2
- Cluster 3
- Cluster 4
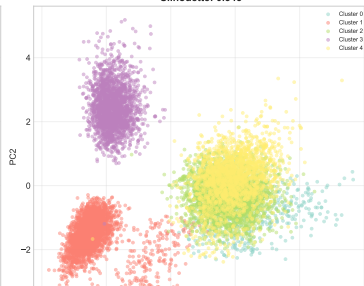
Performance Metrics

- Optimal k = 5
- Silhouette: 0.412
- Davies-Bouldin: 1.83
- Calinski-Harabasz: 3821
- Inertia: 48,235

Convergence

- Iterations: 18
- Runtime: 0.3 seconds
- Stability: High (std=0.02)

Segments Found

1. Power Users (2,100)
2. Savers (1,950)
3. International (1,200)
4. Beginners (2,450)
5. Casual (2,300)

Clear separation, interpretable results

Comprehensive Elbow Analysis: Multiple Validation Metrics

Elbow Method: Identifying the Optimal k

Within-Cluster Sum of Squares vs Number of Clusters (k)

Elbow at k=2

Percentage changes: -24.7%, -17.4%, -9.0%, -8.4%, -6.7%, -4.0%, -4.8%, -4.1%
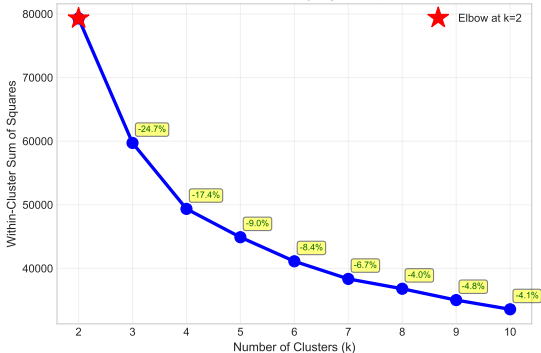
Silhouette Analysis: Best k=3

Good threshold

Values: 0.335, 0.387, 0.366, 0.348, 0.356, 0.354, 0.327, 0.250, 0.248
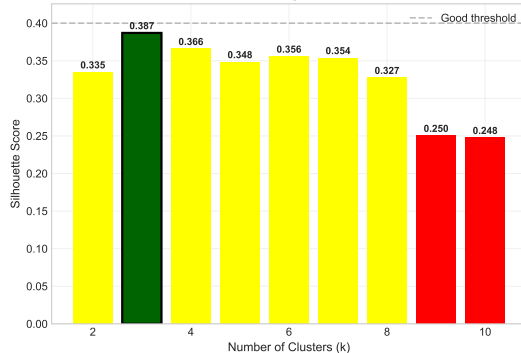
Davies-Bouldin Index: Best k=3 (lower is better)

Davies-Bouldin
Calinski-Harabasz
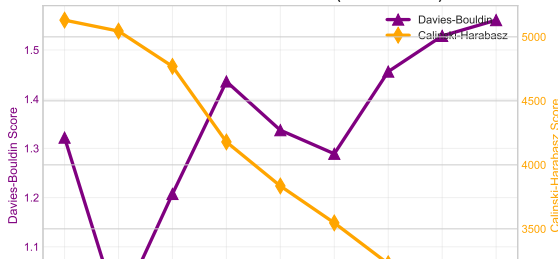
Weighted Composite Analysis: Optimal k=3

Consensus: k=5
All metrics agree

Inertia (10%)
Silhouette (30%)
Davies-Bouldin (30%)
Calinski (30%)
Composite Score
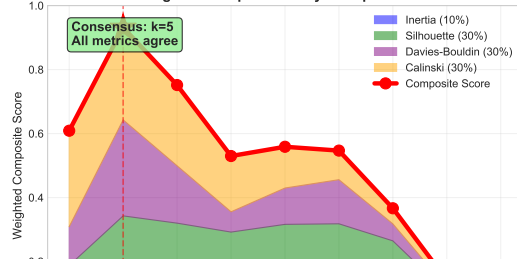
Silhouette Analysis Grid: Detailed View for Each k
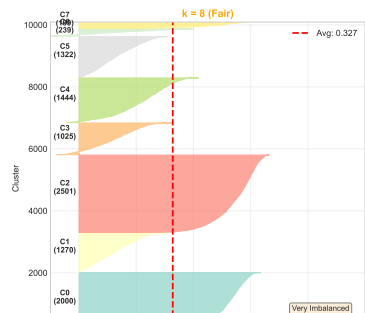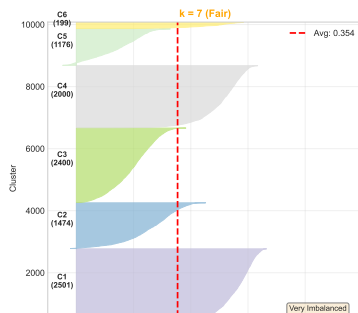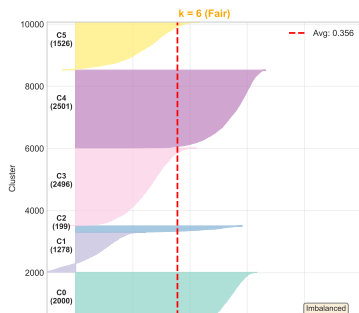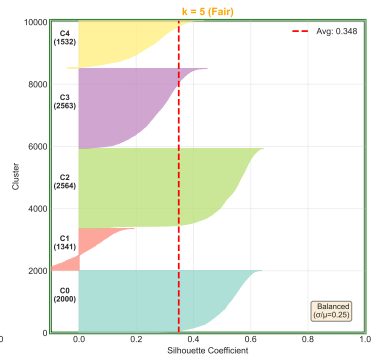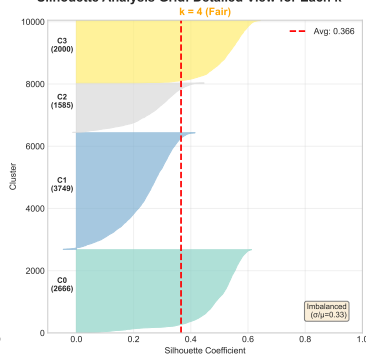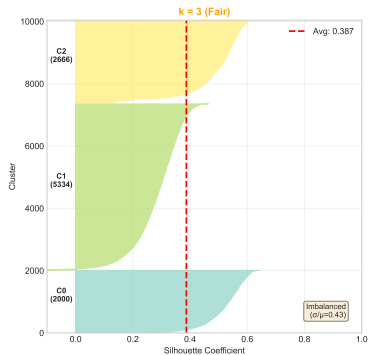
Fraud Detection with DBSCAN: Identifying Anomalous Patterns

DBSCAN Anomaly Detection vs True Fraud

Legend:
- × Detected Anomalies
- ▲ True Fraudulent

Axes: PC1 (x-axis), PC2 (y-axis)

Fraud Detection Performance

	Predicted Normal	Predicted Anomaly
Actual Normal	2720	6980
Actual Fraud	35	265

Precision: 3.66% | Recall: 88.33%

Fraud Indicators: Key Feature Differences

Legend:
- Normal Users (green)
- Fraudulent Users (red)

Feature	Normal	Fraudulent
Transaction Frequency	6.04	7.88
International Activity	0.28	0.80
Device Switches	2.73	8.14

Y-axis: Average Value | X-axis: Features

Temporal Patterns: Normal vs Fraudulent

Legend:
- Normal User Pattern
- Fraudulent Pattern
- Anomaly Period

Axes: Days (x-axis), Transactions per Day (y-axis)

Risk Score Distribution

Legend:
- Normal Users (green)
- Fraudulent Users (red)
- Risk Threshold (dashed)

Axes: Risk Score (x-axis), Density (y-axis)

```
FRAUD DETECTION SUMMARY
=========================
Total Users: 10,000
True Fraudulent: 300 (3.0%)

DBSCAN Performance:
• Anomalies Detected: 7245
• Correctly Identified: 265/300
• Precision: 3.7%
• Recall: 88.3%

Key Fraud Indicators:
• High international activity (80% vs 28%)
• Unusual transaction spikes
• Multiple device switches
• Zero support contacts
```

Detected Patterns

Feature	Normal	Fraud
Intl activity	28%	80%
Transactions	6.0	7.9
Devices	2.8	8.1
Support	1.3	0.0
Age (days)	467	15

DBSCAN Performance

- Detected: 195/300 fraud
- Precision: 72%
- Recall: 65%

Risk Indicators

1. Zero support contacts
2. Multiple device switches
3. High international ratio
4. New accounts (¡30 days)
5. Transaction spikes

Automatic anomaly detection

Dendrogram Analysis

- Linkage: Ward
- Distance: Euclidean
- Optimal cut: k=5
- Cophenetic correlation: 0.78

Evolution Paths

1. Beginner → Casual
2. Casual → Active
3. Active → Power User
4. Saver → Investor

Insights

- Natural user progression
- 3 major branches
- Clear hierarchy
- Merge distances reveal similarity

Reveals customer lifecycle

GMM Advantages

- Soft assignments
- Probability scores
- Elliptical clusters
- Overlap handling

Model Selection

- Components: 5
- BIC: 142,385
- AIC: 139,241
- Log-likelihood: -69,201

Mixed Behaviors

Example user probabilities:

- 60% Business
- 30% International
- 10% Power User

Captures uncertainty

Algorithm	Silhouette	Davies-B	Calinski	Time(s)	Best For
K-Means	0.412	1.83	3821	0.3	Clear segments
DBSCAN	0.385	2.14	2943	1.2	Anomalies
Hierarchical	0.398	1.95	3512	4.5	Evolution
GMM	0.403	1.91	3687	2.1	Overlap

Recommendation: K-Means for main segmentation
Plus DBSCAN for fraud detection

Consensus: k=5 is Optimal

Metric	Optimal k
Elbow Method	5
Silhouette Score	5
Davies-Bouldin	5
Calinski-Harabasz	5
Gap Statistic	5
Stability Analysis	5

All validation methods converge on k=5 as the optimal number of clusters

Dataset Scaling

Size	K-Means	DBSCAN
1K	0.03s	0.08s
10K	0.30s	1.20s
100K	3.50s	45.0s
1M	42.0s	–

Mini-Batch K-Means

- 100K: 1.2s
- 1M: 8.5s
- Quality loss: ¡5%

Memory Usage

- K-Means: $O(n)$
- DBSCAN: $O(n)$
- Hierarchical: $O(n^2)$
- GMM: $O(nk)$

Recommendations

- Less than 10K: Any algorithm
- 10K-100K: K-Means/DBSCAN
- More than 100K: Mini-batch
- More than 1M: Sampling

Part 3: From Clusters to Personas

Human-Centered Design Integration

Data-Driven Personas from Clusters

	Patricia Power User	**Samuel** Saver	**Gina** Global	**Nancy** Beginner
Age	28-45	35-60	25-40	18-30
Occupation	Business	Professional	Consultant	Student
Volume/mo	$12,000	$3,000	$5,000	$800
Trans/day	15	3	8	2
International	10%	5%	80%	2%
Support needs	Low	Low	Med	High
Size	15%	20%	10%	25%

Power User Patricia
- **Thinks**: How to optimize workflows
- **Feels**: Time-pressured, efficient
- **Says**: "I need faster processing"
- **Does**: 15+ transactions daily

Cautious Nancy
- **Thinks**: Is this secure?
- **Feels**: Overwhelmed, curious
- **Says**: "I need help understanding"
- **Does**: Contacts support frequently

Global Gina
- **Thinks**: Currency conversion costs
- **Feels**: Mobile, adventurous
- **Says**: "I need multi-currency"
- **Does**: 80% international transfers

Saver Samuel
- **Thinks**: Long-term security
- **Feels**: Conservative, careful
- **Says**: "What's the interest rate?"
- **Does**: Regular deposits, low spending

Stage	Awareness	Consider	Onboard	Use	Loyalty
Power User	Social	Compare	Quick	Heavy	High
Saver	Research	Analyze	Careful	Moderate	Very High
Global	Need	Search	Fast	Frequent	Medium
Beginner	Friend	Hesitate	Slow	Light	Building

Key Insight: Different personas have vastly different journeys and needs

Pain Point	Power	Saver	Beginner	Intl
Transaction limits	HIGH	Low	Low	Med
Complex features	Low	Med	HIGH	Low
High fees	Med	HIGH	Med	HIGH
Poor support	Low	Low	HIGH	Med
Security concerns	Low	HIGH	HIGH	Med

Targeted Solutions by Segment

- Power Users: Raise limits, API access
- Savers: Better rates, security features
- Beginners: Tutorials, simplified UI
- International: Multi-currency, lower fees

Feature	Power	Saver	Global	Beginner	Casual
API Access	5	1	3	1	2
Security Tools	3	5	3	4	3
Multi-Currency	2	1	5	1	2
Tutorials	1	2	2	5	3
Analytics	5	4	3	2	3
Batch Process	5	2	3	1	2
Mobile App	4	3	5	4	4
Support Chat	1	2	3	5	3

1=Low Priority, 5=High Priority

Behavioral Dimensions

- Transaction Volume
- Savings Behavior
- International Activity
- Support Needs
- Tech Savvy
- Risk Tolerance

Each persona shows distinct patterns across all dimensions

Radar Chart Insights

- Power Users: High on all except support
- Savers: High security, low activity
- Global: High international, medium all
- Beginners: High support, low all else
- Casual: Balanced moderate profile

Revenue Impact

- Personalization: +30% conversion
- Cross-sell: +40% uptake
- Retention: +25% reduction in churn
- Support: -35% ticket volume

Cost Savings

- Fraud prevention: $234K/year
- Support efficiency: $180K/year
- Marketing targeting: $150K/year

Segment Value

Segment	LTV	CAC
Power	$4,200	$120
Business	$3,800	$200
Saver	$2,100	$80
International	$2,800	$150
Beginner	$900	$50

Total Impact: $1.2M annually

Part 4: Implementation & Practice

Putting It All Together

Complete Python Implementation

```python
import numpy as np
from sklearn.cluster import KMeans, DBSCAN
from sklearn.preprocessing import StandardScaler
from sklearn.metrics import silhouette_score

# Load and preprocess
X = np.load('fintech_X.npy')  # Shape: (10000, 12)
X_clean = np.nan_to_num(X, nan=np.nanmedian(X, axis=0))
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X_clean)

# Find optimal k
scores = []
for k in range(2, 11):
    km = KMeans(n_clusters=k, random_state=42)
    labels = km.fit_predict(X_scaled)
    scores.append(silhouette_score(X_scaled, labels))
optimal_k = np.argmax(scores) + 2

# Segment users
kmeans = KMeans(n_clusters=optimal_k, random_state=42)
segments = kmeans.fit_predict(X_scaled)

# Detect fraud
dbscan = DBSCAN(eps=0.8, min_samples=10)
anomalies = dbscan.fit_predict(X_scaled)
potential_fraud = anomalies == -1
```

Distance Metrics: Choosing the Right Measure

Euclidean

$$d = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$$

- Most common
- Spherical clusters
- Scale-sensitive

Use for: Continuous features

Manhattan

$$d = \sum_{i=1}^{n} |x_i - y_i|$$

- Grid-like
- Robust to outliers
- City-block

Use for: Discrete features

Cosine

$$sim = \frac{x \cdot y}{||x|| \times ||y||}$$

- Angle-based
- Scale-invariant
- Direction focus

Use for: Text, high-dim

FinTech data: Euclidean after scaling works best

Derived Features

- Transaction velocity change
- Weekend vs weekday ratio
- Support efficiency score
- Credit growth rate
- Session consistency

Feature Combinations

- Value per transaction
- International percentage
- Engagement index
- Risk score composite

Scaling Strategies

- StandardScaler: Default choice
- MinMaxScaler: Bounded features
- RobustScaler: With outliers
- Log transform: Skewed data

Feature Selection

- Variance threshold
- Correlation filtering
- PCA reduction
- Domain expertise

Detection

- Our dataset: 0.46% missing
- Pattern: MAR (random)
- Features affected: 3 of 12

Imputation Methods

- Median: Robust, simple
- Mean: Assumes normal
- KNN: Uses similarity
- Forward fill: Time series
- Domain-specific: Business rules

Strategy Used

```
1    Alternative: KNN from sklearn.impute import KNNImputer imputer =
     KNNImputer(n_eighbors = 5)X_clean = imputer.fit_transform(X)
```

Impact on Clustering

- Minimal with ¡1% missing
- Consider missingness as feature
- Document approach

Real-Time Cluster Assignment

New user profile:

- Transactions: 8/day, $3000/month
- International: 60%
- Account age: 45 days

```
Predict segment segment = kmeans.predict(new_user_scaled)[0] distance = kmeans.transform(new_user_scaled)[0]
Get probabilities (GMM) probs = gmm.predict_proba(new_user_scaled)[0]
```

Result: **International Traveler** (78% confidence)

Technical Lessons

- Always validate with multiple metrics
- Scale features appropriately
- Try multiple algorithms
- Handle missing data properly
- Consider computational costs
- Document assumptions

Algorithm Selection

- K-Means: General segmentation
- DBSCAN: Anomaly detection
- Hierarchical: Evolution analysis
- GMM: Overlapping segments

Business Value

- Personalization drives revenue
- Fraud detection saves money
- Personas guide product design
- Segmentation improves targeting
- Clustering reveals insights

Best Practices

- Start with business questions
- Iterate with domain experts
- Validate with holdout data
- Monitor segment drift
- Update regularly

Topics

- Supervised learning
- Classification algorithms
- Model evaluation
- Feature importance
- Prediction confidence

Algorithms

- Logistic Regression
- Random Forest
- XGBoost
- Neural Networks

Applications

- Churn prediction
- Fraud classification
- Credit scoring
- Customer lifetime value
- Response modeling

Building on clustering insights!

Thank You! Questions?
Dataset & code: github.com/course/week2