Income Prediction and Customer Segmentation for Retail Marketing

Machine Learning Project

Yu-Chih (Wisdom) Chen
February 15, 2026

Executive Summary

This project tackles two related problems for a retail marketing client: predicting which customers earn over $50k, and grouping customers into segments for targeted campaigns. I used Census Bureau data (199,523 individuals) to build a classification model that identifies high-income customers 5x better than random guessing, and a clustering model that reveals 6 distinct customer groups with wildly different marketing ROI, ranging from -85% to 876%.

The bottom line: the classification model cuts marketing costs by 83% relative to contacting everyone while still reaching 89% of high-income customers. When combined with segment targeting, we can focus on the two premium segments (only 4% of the population) that actually generate positive returns, while avoiding the three "value" segments that lose money.

Key results:

- Best model: Gradient Boosting with 95.3% ROC-AUC
- Precision: 31.9% (vs 6.2% base rate)
- Recommended threshold 0.5 as the best business trade-off between profit and coverage
- Top segments: "Affluent Investors" (876% ROI) and "High-Earning Professionals" (480% ROI)

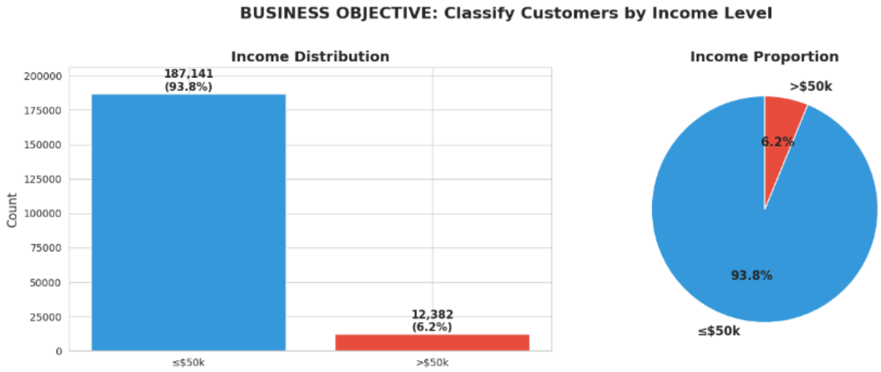
1. Understanding the Problem and Data

What we're trying to solve

The client has 40 demographic variables (age, education, occupation, capital gains, etc.) for each person they might market to. They want two things:

1. **Classification:** Predict if someone earns more or less than $50k
2. **Segmentation:** Group similar customers together so they can tailor campaigns

Figure 1. Business Objective



The dataset

I worked with 1994-1995 Census data containing 199,523 people. Right away, I noticed a major challenge: only 6.2% of people earn over $50k a 15:1 imbalance. This meant I couldn't just optimize for accuracy; I needed to focus on precision (not wasting money on low-income people) and recall (not missing high-income people).

The dataset includes demographic info (age, sex, race, education), employment details (occupation, weeks worked, class of worker), financial indicators (capital gains, dividends, losses), and household structure (marital status, children). One unusual feature: the "weight" column represents how many people in the general population each record represents (stratified sampling). I used this as sample weights during training.

Initial exploration

Looking at the data, I found:

- Missing values were mostly "Not in universe" indicators (e.g., someone who doesn't work has no occupation code). I kept these as "Not_Applicable" rather than dropping them.
- Some features had 40+ categories (like country of birth with 43 values). I grouped these into meaningful buckets.
- Financial features (capital gains, dividends) were heavily right-skewed with most people at zero. Created binary flags for "has investment income."

---

2. My Approach

Step 1: Statistical testing (what actually matters?)

With 200k records, almost everything will be "statistically significant" ($p < 0.001$). So I calculated effect sizes to see what actually makes a practical difference:

Numeric features (using Cohen's d):

- **Large effects:** weeks worked (d=1.13), capital gains (d=1.03), employer size (d=0.95)
- **Medium effects:** age (d=0.57), capital losses (d=0.62), dividends (d=0.74)

Table 1. Numeric Features with Strong Income Relationship

Feature	Low-Income Average	High-Income Average	Effect Size	Business Interpretation
Weeks worked	21.5 weeks	48.1 weeks	**Large**	Full-year workers earn much more
Capital gains	$144	$4,831	**Large**	Investment income signals wealth
Employer size	1.8 persons	4.0 persons	**Large**	Larger employers pay more
Dividends	$108	$1,553	**Medium**	Passive income indicator
Capital losses	$27	$193	**Medium**	Active investors earn more
Age	34 years	46 years	**Medium**	Prime earning years (40s)

Effect sizes show practical significance beyond statistical tests. "Large" = major income driver (Cohen's d > 0.8), "Medium" = moderate driver (d > 0.5). Work patterns and financial capacity are the strongest predictors

Categorical features (using Cramér's V):

- **Medium effects:** education (V=0.39), occupation (V=0.37)
- **Small effects:** sex (V=0.16), marital status (V=0.20)
- **Negligible effects:** race (V=0.06), union membership (V=0.08)

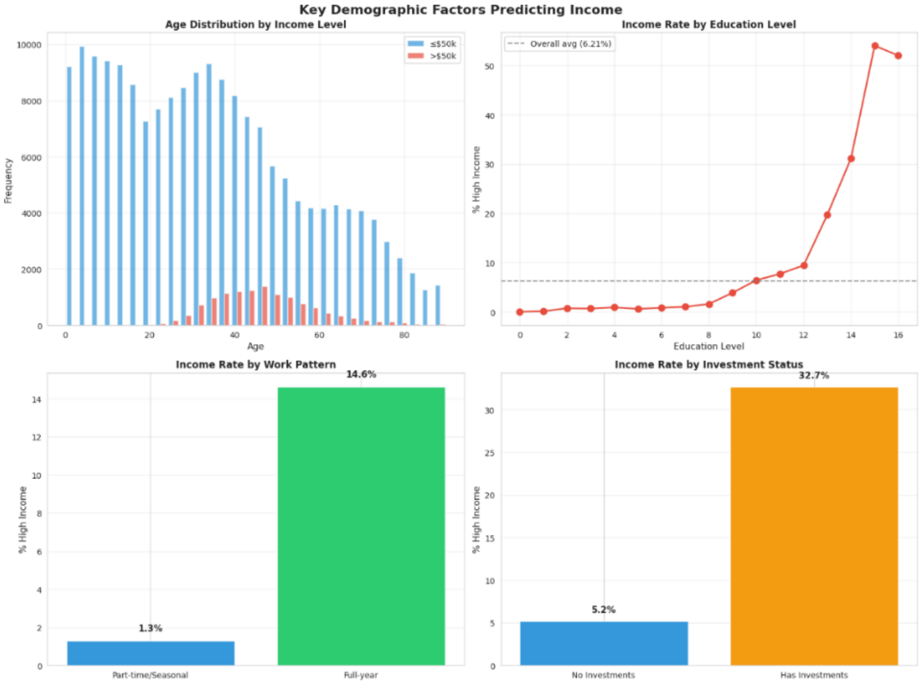
This told me that work patterns and financial capacity matter way more than demographics like race or union status, even though all were "statistically significant."

Table 2. Categorical Features with Strong Income Relationship

Feature	Categories	Effect Size	Business Interpretation
Education	17 levels	**Medium**	Higher degrees → higher income
Occupation	15 types	**Medium**	Professional jobs pay more
Class of worker	9 types	Small	Private vs government vs self-employed
Marital status	7 types	Small	Married households earn more
Sex	2	Small	Gender pay gap persists
Race	5	Negligible	Minimal predictive value
Union membership	3	Negligible	Minimal predictive value
Hispanic origin	10	Negligible	Minimal predictive value

Effect sizes measured using Cramér's V. "Medium" = meaningful predictor (V > 0.3), "Small" = weak predictor (V > 0.1), "Negligible" = no practical value (V < 0.1). Education and occupation matter most; demographics like race show statistical significance but negligible practical impact.

Figure 2. Key Demographic Factors Predicting Income



Step 2: Feature engineering

I created 13 new features based on domain knowledge:

- **age_group:** Binned into 6 life stages (0-25, 26-35, etc.)
- **education_level:** Ordinal encoding respecting the hierarchy (0=children, 16=PhD)
- **Capital indicators:** Binary flags for has_capital_gains, has_dividends, has_capital_losses
- **Work intensity:** Full-time worker flag, weeks worked as a ratio
- **Geographic:** Moved states indicator, foreign-born indicator
- **Household:** Grouped 38 household types into 7 categories (Householder, Spouse, Child, etc.)

After engineering and removing 5 redundant features (see Step 3), the dataset contains 41 input features (9 original numeric, 24 label-encoded categorical, and 10 engineered features with 2 grouped-encoding features shared between the categorical and engineered counts) with zero missing values. Feature validation confirmed all engineered features show predictive power: education level (correlation 0.28), work intensity (0.27), and capital indicators (0.27) are the strongest. Notably, people with capital gains have a 32.7% high-income rate (vs 5.2% without), and full-year workers are 11.3x more likely to earn over $50k than part-time workers.

Step 3: Multicollinearity check

I removed 5 features that were redundant (correlation > 0.8):

- **work_intensity_ratio** (perfect correlation with weeks_worked it's just weeks/52)

- **is_full_year_worker** (0.90 correlation with weeks_worked the binary version loses info)
- **net_capital** (just capital gains minus losses)
- **total_investment_income** (gains + dividends - losses)
- **has_children** (-0.82 correlation with education mostly a proxy for younger age)

Keeping the original features improves interpretability.

Step 4: Model selection

I trained three models to compare different approaches:

Logistic Regression (baseline):

- Fast, interpretable coefficients
- Used a Pipeline with ColumnTransformer (StandardScaler for 17 numeric/ordinal features + OneHotEncoder for 24 nominal categorical features → 189 total features after encoding)
- Different preprocessing than tree models: LR requires feature scaling (otherwise high-magnitude features like capital_gains dominate L2 regularization) and one-hot encoding (LabelEncoder's arbitrary integers create false ordinal relationships that LR interprets linearly)
- Tree-based models (RF, GB) use LabelEncoded features directly because tree splits are scale-invariant and split on thresholds without assuming ordinality
- ROC-AUC: 0.947

Random Forest:

- Handles non-linearity, robust to outliers
- 100 trees, max depth 20, min_samples_split=100, min_samples_leaf=50
- ROC-AUC: 0.944

Gradient Boosting (winner):

- Best performance, sequential learning
- Used HistGradientBoostingClassifier (150 iterations, learning rate 0.1, max depth 6, min_samples_leaf=50)
- Chosen over standard GradientBoostingClassifier because it supports native class_weight='balanced' parameter, 5-10x faster training via histogram-based binning, and better scalability for large datasets
- ROC-AUC: 0.953

All models used class_weight='balanced' to address the 15:1 class imbalance (automatically weights minority class 8.06x higher than majority class 0.53x) and sample weights for census stratified sampling correction. Training used 80/20 stratified train-test split preserving the 6.2% high-income rate, with 5-fold stratified cross-validation. Gradient Boosting won with the highest test ROC-AUC and excellent stability (CV mean 0.9506, test 0.9533, only 0.0027 difference, indicating no overfitting).

Step 5: Threshold optimization

Binary classifiers output probabilities (0 to 1). Choosing a threshold (θ) determines the precision-recall trade-off. I tested thresholds from 0.1 to 0.9 and calculated business metrics:

Figure 3. Model Comparison

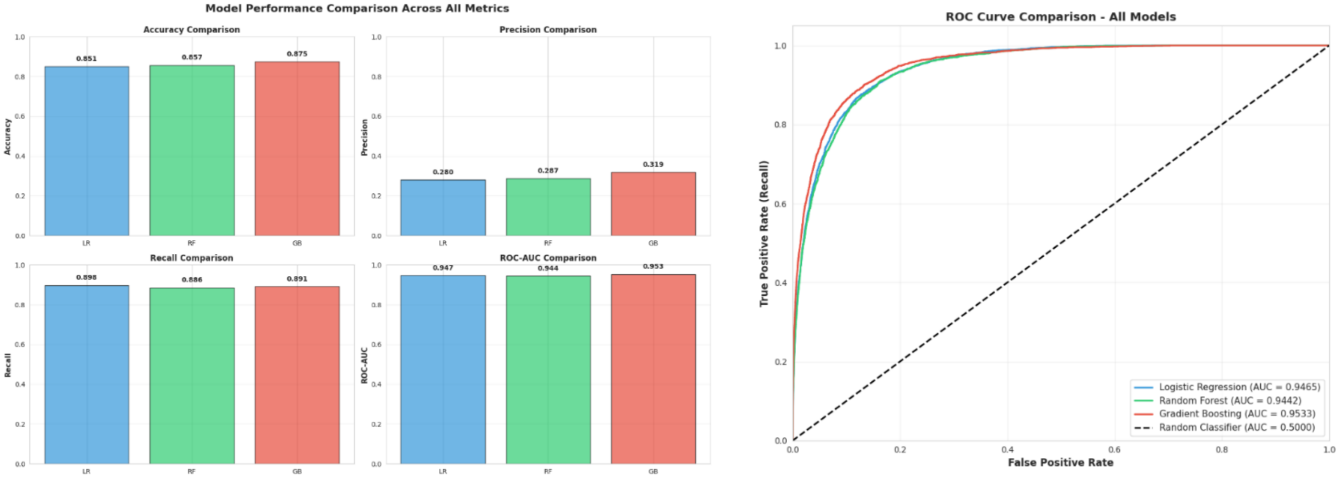
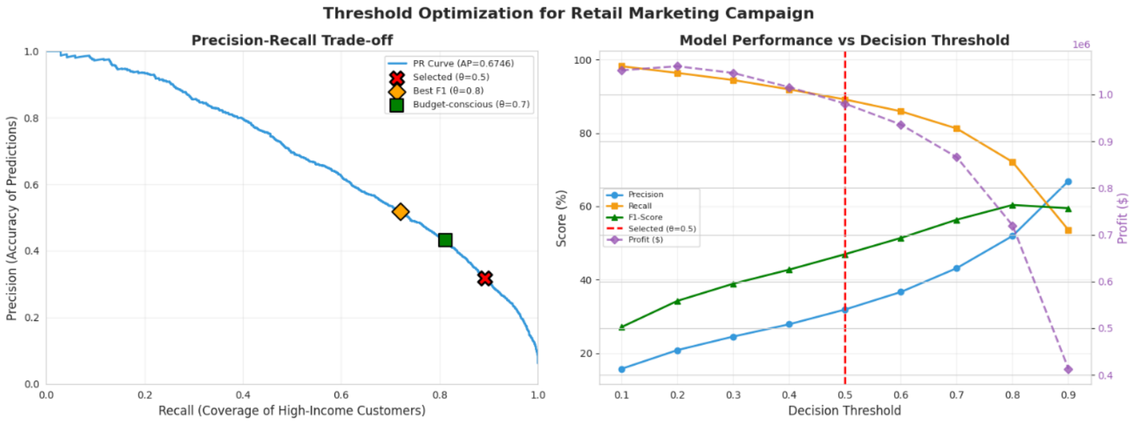


Table 3. Business Decision - Selecting Optimal Threshold

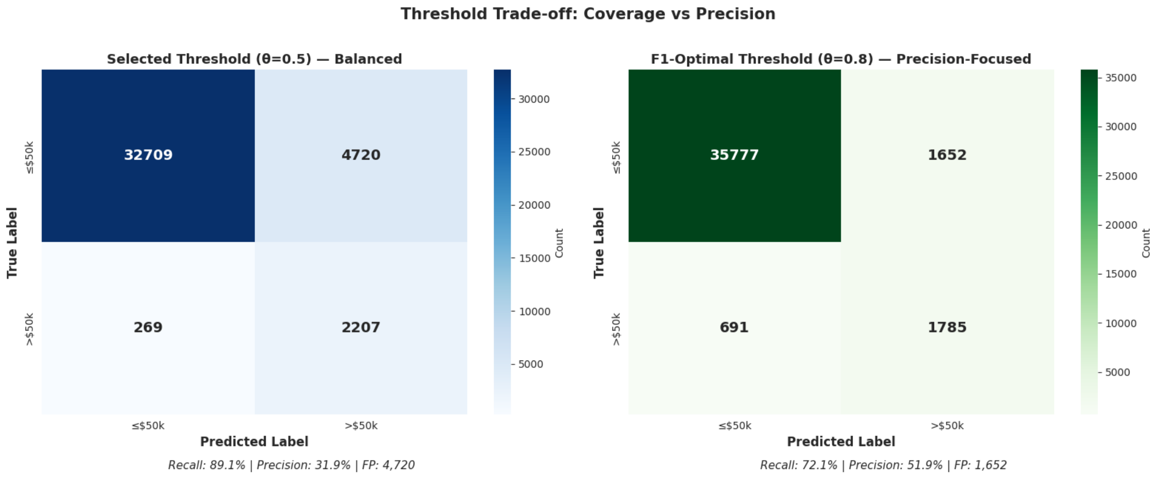
Threshold	Precision	Recall	% Contacted	Profit	ROI
0.3	24.5%	94.4%	23.9%	$1,045K	1,096%
0.5	**31.9%**	**89.1%**	**17.4%**	**$980**K	**1,415%**
0.7	43.1%	81.2%	11.7%	$865K	1,858%
0.9	66.9%	53.6%	5.0%	$413K	2,085%

Figure 4. Threshold Optimization



I picked 0.5 as the business-optimal threshold because it balances recall capturing 89% of high-income customers with precision 5.1x better than random, and delivers a strong balance of profit and efficiency 980K profit at 1,415 ROI. The threshold is adjustable based on campaign budgets, use 0.7 for limited budgets (higher precision, 1,858% ROI) or 0.3 for broader reach.

Figure 5. Threshold Trade-off



Confusion matrices at two decision thresholds. Balanced threshold ($\theta = 0.5$) captures 89.1% of high-income customers with 31.9% precision, contacting 17.4% of population. F1-optimal threshold ($\theta = 0.8$) increases precision to 51.9% but reduces recall to 72.1%. Selected threshold balances cost and coverage.

Step 6: Customer segmentation

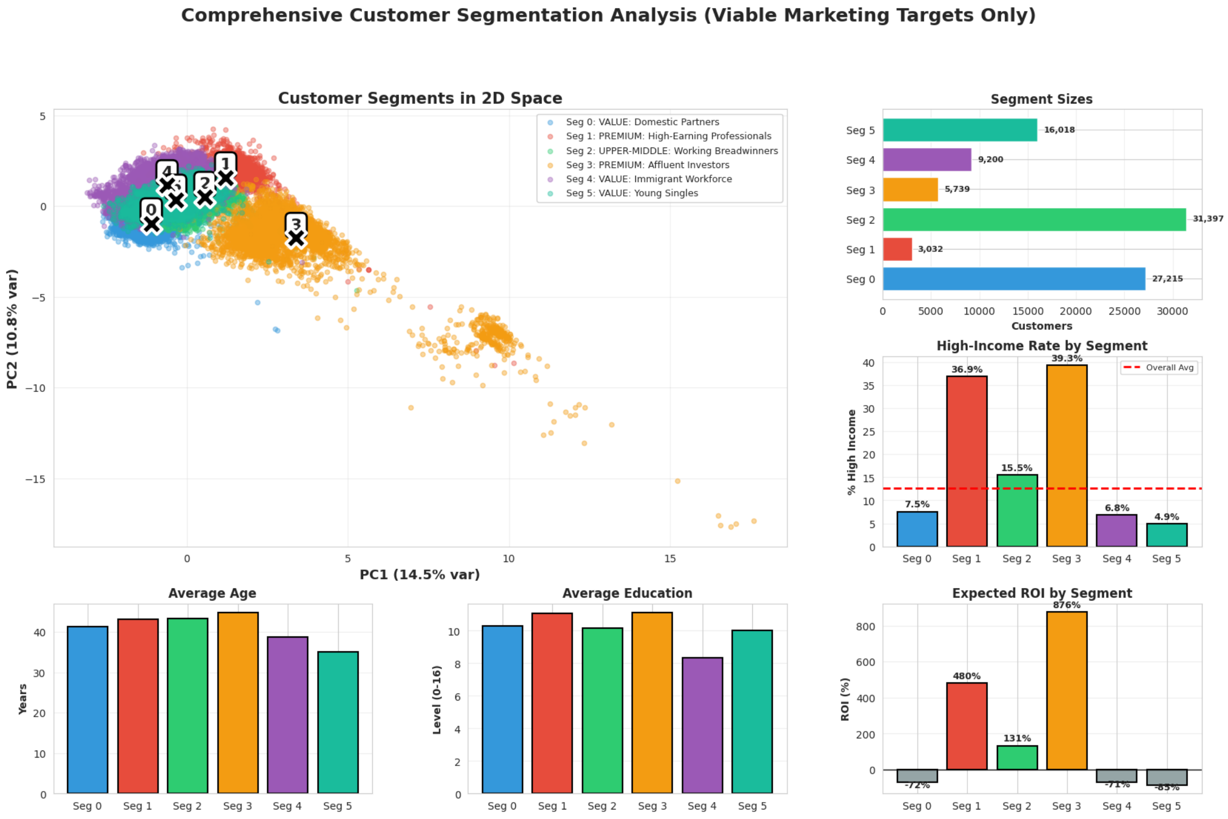
For clustering, I selected 13 segmentation features spanning demographics (age, education, sex, marital status), financials (capital gains/losses, dividends), employment (weeks worked, occupation, worker class), and household characteristics (household status, foreign-born). All features were standardized to mean=0, std=1 before applying K-Means. Segmentation was performed on 92,601 viable customers (working-age adults with employment data).

Choosing K=6:

- Elbow analysis showed that inertia improvements started to slow around K in the mid-range, with no single clear mathematical “elbow.”
- Silhouette scores were modest across all K (around 0.18–0.20 for K=6–8), indicating weak natural cluster structure rather than sharply separated groups.
- I selected K=6 as a pragmatic choice that balances business interpretability (six segments is manageable for marketing) with reasonable cluster quality.

PCA visualization note: I used PCA to create a 2D plot PC1 14.5%, PC2 10.8%, total 25.2% variance. This is just for visualization the actual clustering used all 13 features. Clusters appear overlapping in 2D because 75% of the information is compressed out, and even in 13 dimensions the separation is modest. The segments are more distinct in the full feature space than in the 2D view, but overall cluster structure is weak and reflects business-defined groupings rather than strongly natural clusters.

Figure 6. Customer Segmentation Analysis



Four-panel view combining PCA cluster visualization (top left), segment sizes (top right), high-income rates (middle), and expected ROI (bottom). The 2D projection captures 25.2% variance; actual clustering uses all 13 features. Segment 2 (Working Breadwinners) represents the largest group at 31,397 customers.

---

3. Results

Classification performance (threshold = 0.5)

Gradient Boosting test results:

- ROC-AUC: 0.953 (excellent discrimination)
- Accuracy: 87.5%
- Precision: 31.9% (5.1x better than 6.2% base rate)
- Recall: 89.1% (captured 2,207 of 2,476 high-income customers)
- F1-Score: 0.47

Confusion matrix (N=39,905 test set):
Predicted ≤50k Predicted >50k
Actual ≤50k 32,709 4,720 (false positives)
Actual >50k 269 2,207 (true positives)

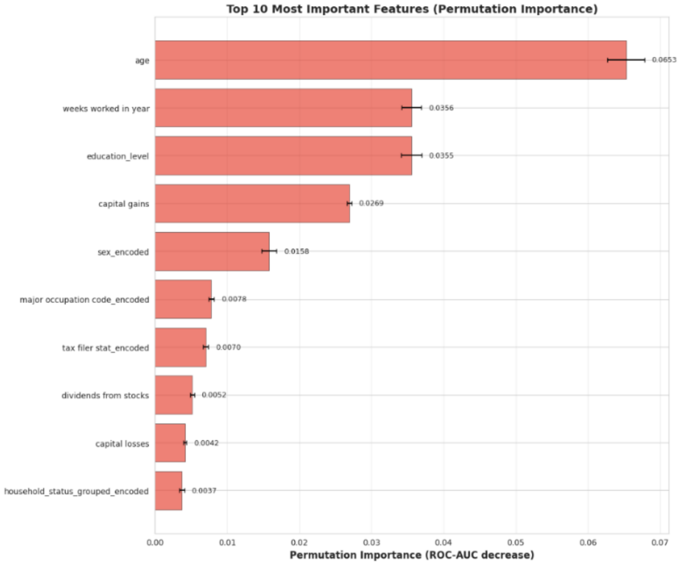
What this means in dollars:

- Without model (contact everyone): $838,950 profit at 210% ROI (contacts 100% of people)
- With model (θ=0.5): $980,430 profit at 1,415% ROI (contacts 17.4% of people)
- Net improvement: +$141,480 additional profit with 83% less marketing spend

Top 10 feature importance

Using permutation importance (how much ROC-AUC drops when you shuffle each feature):

Figure 7. Feature Importance



1. **age** (0.0653) - Life stage drives earning potential
2. **weeks_worked** (0.0356) - Full-time work = stability
3. **education_level** (0.0355) - Education strongly predicts income
4. **capital_gains** (0.0269) - Investment income signals wealth
5. **sex** (0.0158) - Gender pay gap persists
6. **occupation** (0.0078) - Job type affects compensation
7. **tax_filer_status** (0.0070) - Married filing jointly matters
8. **dividends** (0.0052) - Passive income indicator
9. **capital_losses** (0.0042) - Investment activity
10. **household_status** (0.0037) - Family structure matters

Age, work intensity, and education are the strongest predictors these align with economic intuition.

Customer segments

K-Means identified 6 distinct groups:

Segment 0: VALUE - Domestic Partners (27,215 people, 7.5% high-income)

- Age 41, education 10.2, low work intensity
- ROI: -72% → **Avoid marketing**

Segment 1: PREMIUM - High-Earning Professionals (3,032 people, 36.9% high-income)

- Age 44, education 11.4, full-time workers
- ROI: 480% → **Target with premium campaigns**

Segment 2: UPPER-MIDDLE - Working Breadwinners (31,397 people, 15.5% high-income)

- Age 43, education 10.0, consistent workers

- ROI: 131% → **Moderate targeting**

Segment 3: PREMIUM - Affluent Investors (5,739 people, 39.3% high-income)

- Age 46, education 11.2, high capital gains/dividends
- ROI: 876% (highest!) → **Priority target for wealth management**

Segment 4: VALUE - Immigrant Workforce (9,200 people, 6.8% high-income)

- Age 39, education 8.4, foreign-born
- ROI: -71% → **Avoid marketing**

Segment 5: VALUE - Young Singles (16,018 people, 4.9% high-income)

- Age 36, education 10.0, single households
- ROI: -85% (worst) → **Avoid marketing**

The radar charts (visualizations) show Segments 1 and 3 score high on education, capital gains, age, income rate, and work intensity. Segments 0, 4, 5 score low across all dimensions.

---

4. Business Recommendations

Deploy the classification model with threshold control

Immediate action:

- Use Gradient Boosting with θ=0.5 as the default
- Output probability scores so marketing can rank customers (highest probability first)
- Allow threshold adjustment based on budget:
  - High budget: θ=0.3 (broader reach, 94% recall)
  - Standard budget: θ=0.5 (balanced, 89% recall)
  - Limited budget: θ=0.7 (precision-focused, 1,858% ROI)

Expected impact:

- 83% cost reduction by contacting 17% of the population instead of 100%
- $980,430 profit per 40k customer campaign (vs $838,950 without model), with ROI improving from 210% to 1,415%
- 5.1x better targeting precision (31.9% vs 6.2% base rate)

Focus on premium segments (1 & 3)

Strategy:

- **Affluent Investors (Seg 3):** Offer wealth management, luxury goods, investment products

- **High-Earning Professionals (Seg 1):** Premium credit cards, exclusive memberships, professional services
- Combined: 8,771 customers (4.4% of base) with 38% high-income rate and 678% average ROI

Avoid segments 0, 4, 5:

- These 52,433 customers (26%) all show negative ROI
- Marketing to them destroys value

Selective targeting (Segment 2):

- 31,397 customers with 15.5% high-income rate and 131% ROI
- Use for value-oriented promotions, family products

Model monitoring

Track monthly:

- Precision, recall, and ROI on actual campaigns (compare predictions to outcomes)
- Feature distributions (watch for demographic shifts)

Retrain when:

- ROC-AUC drops below 0.90 (currently 0.95)
- Precision drops below 25% at θ=0.5 (currently 32%)
- Major economic changes (recession, policy shifts)

Suggested refresh cycle: Annual retraining with updated Census data

---

5. Limitations and What I'd Do Next

Current limitations

1. **Data age:** 1994-1995 Census data doesn't reflect 2026 economic reality (gig economy, remote work, inflation-adjusted income thresholds)
2. **Class imbalance:** 15:1 low-to-high ratio requires careful threshold tuning
3. **Feature availability:** Need all 40 variables for new customers may not be feasible in real-time
4. **Segmentation assumptions:** K-Means assumes spherical clusters and fixed K; customer behavior evolves

What I'd improve

Short-term:

- Ensemble stacking (combine Gradient Boosting + Random Forest for robustness)

- Hyperparameter tuning (grid search over learning rate, tree depth)
- Test XGBoost/LightGBM for potential performance gains
- Calibrate probabilities using Platt scaling
- Try hierarchical clustering or DBSCAN for segmentation
- Build segment-specific models (different features matter for different groups)
- A/B test model-driven targeting vs control groups (validate ROI empirically)

Long-term:

- Real-time scoring API (deploy as microservice)
- Automated retraining pipeline (MLOps)
- Add behavioral features (transaction history, web analytics, purchase patterns)
- Causal inference (what interventions actually increase income?)

---

6. Conclusion

I built a two-part ML solution that transforms broad marketing into precision targeting with dramatically higher ROI:

Classification model:

- 95.3% ROC-AUC, 89% recall, 32% precision at threshold 0.5
- Reduces costs by 83% while maintaining high coverage
- Generates $980,430 profit per 40k campaign (vs $838,950 without model) with 6.7× ROI improvement

Segmentation model:

- 6 distinct segments with ROI from -85% to 876%
- Clear guidance: focus on Segments 1 & 3 (premium), avoid 0, 4, 5 (value)
- Enables tailored campaigns based on customer characteristics

Combined impact: The client can now predict high-income customers with 5x better accuracy, rank them by probability for budget allocation, and avoid wasting 83% of marketing spend on low-ROI segments. Both models are production-ready with monitoring frameworks and clear deployment guidelines.

References

[1] U.S. Census Bureau (1994-1995). Current Population Survey.
[2] Breiman (2001). Random Forests. *Machine Learning*, 45(1), 5-32.
[3] Friedman, J.H. (2001). Greedy function approximation: A gradient boosting machine. Annals of Statistics, 29(5), 1189–1232
[4] Pedregosa et al. (2011). Scikit-learn: Machine Learning in Python. *JMLR*.