Income Classification & Customer Segmentation Project Report

Based on 1994–1995 U.S. Census Bureau Population Survey Data

Prepared By: **Clement T. Okolo**
Client: Retail Business

TABLE OF CONTENT

1. Executive Summary **3**
2. Data Exploration **3**
2.1 Dataset Overview 3
2.2 Key Findings 4
3. Data Preprocessing **4**
3.1 Feature Engineering & Selection 4
3.2 Handling Missing Values & Placeholders 5
3.3 Categorical Encoding 5
3.4 Feature Scaling 5
3.5 Dataset Split 5
4. Classification Model **6**
4.1 Model Architecture 6
4.2 Training Procedure 6
4.3 Evaluation Results 6
4.4 Best Model Deep Dive: XGBoost 6
4.5 Overfitting Analysis 7
5. Segmentation Model **7**
5.1 Approach & Algorithm Selection 7
5.2 Optimal Cluster Selection 8
5.3 Cluster Profiles & Marketing Interpretation 8
5.4 Cluster Validation 9
6. Model Usage Recommendation **9**
6.1 How to Use the Classification Model 9
6.2 How to Use the Segmentation Model 9
6.3 Combined Model Strategy 10
7. Improvement Opportunities **10**
7.1 Classification Model 10
7.2 Segmentation Model 10
References **10**

1. Executive Summary

This report presents a comprehensive machine learning solution developed to support the retail client's marketing initiatives. Two complementary models were built using 199,523 observations drawn from the 1994–1995 U.S. Census Bureau Current Population Survey:

- A binary income classification model that predicts whether an individual earns below or above $50,000 annually, enabling income-based audience targeting.
- A K-means customer segmentation model that partitions the population into four behaviorally and demographically distinct groups, each with its own marketing profile.

The best-performing classifier, XGBoost, achieved 95.7% accuracy and a ROC-AUC of 0.953 on the test set, demonstrating strong discriminative power for income prediction. The segmentation model identified four clearly differentiated market segments: a niche newborn segment, a junior consumer group, a middle market, and a core high-income workforce.

> ***Model Usage**: I recommend the use of the classification model to pre-screen the marketing list by income, then apply the segmentation model to tailor creative messaging and channel strategy to each customer group.*

Code: https://github.com/Clement-Okolo/Census-Bureau-Project/tree/main

2. Data Exploration

2.1 Dataset Overview

The dataset contains 199,523 records across 42 columns: 40 demographic and employment-related predictor variables, one survey weight column, and one binary income label. Variables span a wide range of domains including age, education, marital status, occupation, industry classification, country of birth, and capital income.

Property	Value
Total Records	199,523
Total Features (raw)	42 (incl. weight & label)
Numerical Features	7
Categorical Features	33
Target Variable	'- 50000.' or '50000+.'
Survey Years	1994 and 1995
Class Balance	~93.8% low income vs. ~6.2% high income

2.2 Key Findings

Class Imbalance

The dataset exhibits a severe class imbalance: approximately 93.8% of records are labeled as low income (< $50k) and only 6.2% as high income (> $50k). This imbalance directly affects model training, as a naive classifier predicting "low income" for every observation would achieve ~94% accuracy while being entirely useless for marketing high-income audiences. This guided our choice of ROC-AUC and per-class recall as the primary evaluation metrics rather than raw accuracy.

Missing and Placeholder Values

The variable 'hispanic origin' contained true null values, which were later imputed with the median. Several other categorical variables used the string 'Not in universe' as a placeholder, indicating the question was not applicable to the respondent rather than a missing response. Additionally, the string '?' appeared in geographic migration variables (e.g., 'migration code-change in msa', 'country of birth father'). Both placeholders were handled during the data preparation step.

Numerical Variable Distributions

Key numerical features exhibit strong right-skewness. The variables 'wage per hour', 'capital gains', and 'dividends from stocks' all had a median of zero, indicating that the majority of the population has no capital income. Extreme values (capped at 99,999) appeared in these fields, suggesting data censoring. The 'age' variable ranged from 0 to 90, with a mean of approximately 34, pointing to a highly diverse demographic sample.

Feature Correlations

A Pearson correlation analysis on numerical variables revealed that 'weeks worked in year' is highly correlated with both 'detailed industry recode' ($r = 0.75$) and 'num persons worked for employer' ($r = 0.75$), suggesting potential multicollinearity. The 'year' variable showed near-zero correlation with all other features including the label, indicating it carries no predictive power.

3. Data Preprocessing

3.1 Feature Engineering & Selection

The following decisions were made when preparing features for both models:

- **Weight column removed**: This survey metadata variable represents the population distribution implied by stratified sampling and is not a demographic characteristic.
- **Year variable removed:** Encodes only the survey collection year (1994/1995) and carries no meaningful predictive or segmentation signal.

- **Weeks worked in year removed (classification only):** Highly correlated with industry recode (r = 0.75), removed to reduce multicollinearity in the classifier. Retained for the segmentation model as it provides a distinct temporal labor dimension.

- **Label removed for segmentation**: Including the income label in clustering would artificially bias segments toward income differences rather than allowing natural demographic groupings to emerge.

3.2 Handling Missing Values & Placeholders

- **NaN in 'hispanic origin':** Handled with the column median, a robust choice that is not distorted by extreme values.
- **'?' placeholders:** In the classification model, these were dropped via 'drop_first' one-hot encoding to avoid multicollinearity. In the segmentation model, they were retained as their own binary feature to capture individuals with incomplete geographic profiles, a potentially meaningful 'niche' segment.
- **'Not in universe' placeholders:** Retained as a one-hot category in both models, as this indicates question non-applicability (e.g., not enrolled in education), which is a substantive characteristic of the individual.

3.3 Categorical Encoding

All categorical features were one-hot encoded using pandas 'get_dummies'. The classification model used 'drop_first=True' to avoid perfect multicollinearity (dummy variable trap). The segmentation model used 'drop_first=False' to retain full category information for clustering. After encoding, the classification feature matrix expanded to 377 columns, and the segmentation matrix to approximately 396 columns.

> ***Question for Client:*** *Do we have a data dictionary that formally defines the meaning of 'Not in universe' and other placeholder values? This information would allow for more informed decisions on whether to retain or remove these placeholders, which could potentially improve model performance.*

3.4 Feature Scaling

StandardScaler was applied to normalize all features to zero mean and unit variance. This is critical for distance-based algorithms (SVM, K-Means) and regularized models (Logistic Regression) that are sensitive to feature magnitude. Gaussian Naive Bayes was trained on unscaled features because it assumes feature independence with its own distributional parameters.

3.5 Dataset Split

An 80/20 stratified train-test split was used for the classification task, resulting in 159,618 training records and 39,905 test records. Stratification was applied to ensure both splits preserve the 93.8%/6.2% class ratio, preventing misleadingly optimistic test set metrics.

4. Classification Model

4.1 Model Architecture

Six candidate classifiers spanning a wide complexity spectrum were evaluated to identify the model that best balances predictive performance with interpretability and generalization:

Algorithm	Rationale
Logistic Regression	Linear baseline; interpretable coefficients for business stakeholders
Random Forest	Robust ensemble; handles non-linearities and mixed feature types
Gradient Boosting	Sequential boosting for high accuracy on tabular data
XGBoost	State-of-the-art gradient boosting with regularization; fast at scale
Gaussian Naive Bayes	Probabilistic baseline; fast but assumes feature independence
SGD-SVM (modified huber)	Scalable linear SVM approximation for large datasets

4.2 Training Procedure

Each model was trained on the scaled training set (except Naive Bayes, which used unscaled features). Default hyperparameters were used as a first pass; all tree-based models used 100 estimators, and Logistic Regression was allowed up to 1,000 iterations for convergence. XGBoost used log-loss as its evaluation metric. Cross-validation and hyperparameter tuning were identified as improvement opportunities but were not applied in this initial iteration.

4.3 Evaluation Results

The following table shows test set performance for all six candidate models:

Model	Accuracy	Precision	Recall	F1-Score	ROC-AUC
XGBoost	95.71%	95.42%	95.71%	95.48%	0.9530
Random Forest	95.60%	95.22%	95.60%	95.32%	0.9510
Gradient Boosting	94.87%	94.51%	94.87%	94.62%	0.9480
Logistic Regression	93.40%	92.97%	93.40%	93.07%	0.9120
SGD-SVM	93.10%	92.77%	93.10%	92.88%	0.9050
Gaussian Naive Bayes	80.20%	88.12%	80.20%	81.45%	0.8240

XGBoost is the best ROC-AUC (0.9530), best accuracy (95.71%), and best weighted F1-score. Its built-in regularization (L1/L2) also mitigates overfitting risk compared to the near-perfect Random Forest training scores.

4.4 Best Model Deep Dive: XGBoost

While aggregate metrics are strong, the per-class breakdown reveals an important asymmetry with direct marketing implications:

Class	Precision	Recall	F1-Score	Support
< $50,000 (Low Income)	97%	99%	98%	~37,400
> $50,000 (High Income)	75%	46%	57%	~2,500

The model is highly conservative in predicting high-income earners: it correctly identifies 99% of low-income individuals but misses 54% of actual high earners (recall = 0.46). When it does flag someone as high income, it is correct 75% of the time. This asymmetry is a direct consequence of the class imbalance in training data.

Business Implication: If the goal is to maximize the number of high-income earners reached by marketing campaigns (coverage), the current model will significantly undercount this group. Reducing the classification threshold below the default 0.5 would increase recall at the cost of precision (more marketing spend on misclassified low-income individuals). The optimal threshold should be calibrated based on the relative cost of a False Negative (missed high-income customer) versus a False Positive (wasted marketing impression).

> ***Questions for Client:*** *What is the relative business cost of missing a high-income customer versus incorrectly targeting a low-income customer? This will determine whether we should tune the probability threshold to maximize recall, precision, or F1 for the high-income class.*

4.5 Overfitting Analysis

Training set scores were notably higher than test set scores—most dramatically for Random Forest (near 99% training accuracy vs. ~95.6% test accuracy). XGBoost also showed a gap but a smaller one, suggesting its built-in regularization provides better generalization. Logistic Regression showed the smallest train-test gap, confirming it as the least-overfit model, though at the cost of predictive performance.

5. Segmentation Model

5.1 Approach & Algorithm Selection

K-Means clustering was selected as the primary segmentation algorithm for the following reasons:

- **Scalability:** Efficiently handles the 199,523-row, ~396-column feature matrix.
- **Interpretability:** Cluster centroids provide intuitive, human-readable segment profiles for business stakeholders.

- **Configurability:** The number of clusters K is a controllable business parameter that can be tuned to match the client's operational capacity for distinct marketing campaigns.

The k-means++ initialization strategy was used (init='k-means++') to improve centroid initialization quality and reduce sensitivity to random seed selection. Each model run used n_init=10 independent initializations to select the best solution by inertia.

5.2 Optimal Cluster Selection

The Elbow Method and Silhouette Score analysis were used jointly to determine the optimal number of clusters K, evaluated over the range K=2 to K=10:

- **Elbow Method:** A clear inflection point appeared at K=5, where the rate of WCSS decrease slows substantially, suggesting diminishing returns from additional clusters.
- **Silhouette Score:** While K=2 produced the highest silhouette score, K=4 offered a meaningfully higher score than K=5 while providing substantially richer market structure than a simple binary split.

Decision: K=4 was selected as the optimal number of clusters. It balances statistical cluster quality (second-best silhouette score) with business interpretability (four actionable, differentiated segments rather than two).

5.3 Cluster Profiles & Marketing Interpretation

The final K-Means model (K=4) produced the following segment distribution and key characteristics:

Segment	Population	Share	Avg Age	High Income %	Avg Weeks Worked	Marketing Label
Cluster 0	~61,500	30.8%	45.9 yrs	~28%	31.1 wks	Middle Market
Cluster 1	~73,200	36.7%	44.2 yrs	~32%	31.6 wks	Core Workforce
Cluster 2	~15,200	7.6%	~0 yrs	~0%	~0 wks	Newborn Niche
Cluster 3	~63,600	31.8%	13.1 yrs	~0%	~0 wks	Junior Consumer

Cluster 1: Core Workforce / Primary Market (36.7%)

The largest and highest-value segment. Adults averaging 44.2 years of age with the highest income potential (~32% earning above $50k) and strong labor market participation. This segment represents the retail client's primary audience for premium product lines, loyalty programs, and financial services cross-promotion. Marketing should emphasize digital given their engagement and purchasing power.

Cluster 0: Middle Market (30.8%)

Slightly older adults (avg. 45.9 years) with slightly lower income share (~28% high earners) but still strong retail spending power. This group likely represents middle-career professionals and dual-income households. Marketing strategy should emphasize value, reliability, and family-oriented products. Promotional offers and loyalty discounts may resonate strongly with this segment.

Cluster 3: Junior Consumer Market (31.8%)

Children and young teenagers averaging 13.1 years of age with negligible independent income. However, they are highly relevant as household influencers. Marketing to parents in Clusters 0 and 1 about youth products, back-to-school categories, and entertainment should leverage this segment's preferences. Direct digital channels targeting parents with children in this age range would be effective.

Cluster 2: Newborn Niche (7.6%)

A tightly isolated small population of infants (avg. age ~0) that the model successfully identified as a distinct household life stage. While these individuals do not make purchasing decisions, households with newborns represent high-value retail targets for baby products, parental leave shopping, and household formation goods. This insight provides actionable intelligence for lifecycle marketing.

5.4 Cluster Validation

PCA (Principal Component Analysis) was applied to reduce the ~396-dimensional feature space to 2 dimensions for visualization. The resulting scatter plot confirmed clear spatial separation between the four cluster centroids, particularly for Clusters 0, 1, and 3. Cluster 2 (Newborn Niche) formed a very compact, isolated region, confirming that the model successfully isolated a highly homogeneous age-based subgroup. The 'band' structure visible in the PCA plot suggests segments are strongly defined by categorical characteristics rather than continuous variables alone.

6. Model Usage Recommendation

6.1 How to Use the Classification Model

The income classifier should be integrated into the client's marketing list acquisition and pre-screening workflow. Given the current model's behavior (high precision, low recall for high earners), we recommend the following deployment approach:

- **Conservative use case:** Use a 0.5 default threshold to identify a high-confidence pool of predicted high-income individuals for premium product campaigns.
- **Coverage-optimized use case:** Lower the classification threshold to 0.3–0.4 to maximize recall for the high-income class, accepting a higher rate of low-income individuals in the campaign pool.
- **Cost-benefit threshold tuning:** Conduct A/B testing to measure the revenue difference between marketing to predicted-high-income individuals versus the broader population, then calibrate the threshold accordingly.

6.2 How to Use the Segmentation Model

The four-cluster segmentation model enables distinct marketing strategies for each identified group:

Segment	Recommended Marketing Strategy	Suggested Channels
Core Workforce (Cluster 1)	Premium products, financial services, loyalty programs	Email, Paid Search, LinkedIn
Middle Market (Cluster 0)	Value-driven promotions, family products, reliability messaging	Email, Facebook, Circulars
Junior Consumers (Cluster 3)	Youth products via parents; school season promotions	Parent-targeted social, Display
Newborn Niche (Cluster 2)	Baby essentials, household formation, parenting content	Parenting websites, Display

6.3 Combined Model Strategy

The most powerful deployment combines both models: use the classifier to score each individual's probability of being a high-income earner, then overlay the segmentation labels to identify which cluster they belong to. This two-dimensional matrix produces highly targeted, personalized marketing audiences, for example: 'high-probability high-income Core Workforce' individuals are likely the most valuable marketing audience for premium retail products.

7. Improvement Opportunities

7.1 Classification Model

- **Class imbalance handling:** Explore SMOTE oversampling of the minority (high income) class to improve recall for high earners.
- **Neural network comparison:** Evaluate FT-Transformer or TabNet to confirm XGBoost is not outperformed by deep learning approaches on this tabular dataset.

7.2 Segmentation Model

- **K-Prototypes algorithm:** Replace the one-hot encoding + K-Means pipeline with K-Prototypes, which natively handles mixed numerical/categorical data. This would eliminate the ~400-column feature explosion that currently dilutes the silhouette score.
- **Sub-segmentation of Core Workforce:** Run a second K-Means pass exclusively on Clusters 0 and 1 to identify finer distinctions within the adult working population.

References

- Olamendy, J. C. (2025). *SGDClassifier: The Powerhouse for Large-Scale Classification*. Medium.
- Prakash, P. (2023). *Understanding Baseline Models in Machine Learning: Importance, Strategies, and Application to Imbalanced Classes*. Medium.
- Chen, T., & Guestrin, C. (2016). XGBoost: A Scalable Tree Boosting System. Proceedings of KDD '16.