CENSUS BUREAU INCOME CLASSIFICATION AND CUSTOMER SEGMENTATION PROJECT REPORT FOR CLIENT

Varun Satheesh 12/2025

1. EXECUTIVE SUMMARY

==

This project implements two machine learning models for a retail business client using Census Bureau data:

1. Classification Model: Predicts whether a person earns more than $50,000 or less than/equal to $50,000
2. Segmentation Model: Creates customer segments for targeted marketing

The dataset contains 199,523 observations with 40 demographic and employment variables. The classification model achieved 90.7% accuracy with a ROC-AUC score of 0.947, while the segmentation model identified distinct customer segments for marketing purposes.

2. DATA EXPLORATION AND PRE-PROCESSING APPROACHES

==

2.1 Data Overview

- Dataset Size: 199,523 observations with 40 features
- Target Variable: Income label (- 50000. for <=$50k, 50000+. for >$50k)
- Class Distribution: Highly imbalanced (93.8% <=$50k, 6.2% >$50k)
- Data Source: Census Bureau survey data

2.2 Data Exploration Findings

- Missing Values: Found '?' placeholders and NaN values in both numerical and categorical columns
- Feature Types: Mix of numerical (age, capital gains/losses, weeks worked, wage per hour) and categorical features (education, marital status, race, sex, occupation, industry)
- Non-Predictive Features: Identified 'weight' (population representation weights) and 'year' (survey year) as non-predictive features that should be excluded from modeling

2.3 Pre-Processing Steps

The following pre-processing pipeline was implemented:

a) Data Cleaning:
   - Removed trailing spaces and periods from label column
   - Identified and handled missing values ('?' and NaN)
   - Separated target variable from features

b) Feature Selection:
   - Removed 'weight' column (population weights, not predictive)
   - Removed 'year' column (survey year, not useful for prediction)
   - Retained all other demographic and employment features

c) Missing Value Treatment:
   - Numerical Features: Replaced '?' with NaN, then filled with median values

- Categorical Features: Replaced missing values with 'Unknown' category

d) Feature Encoding:
- Categorical Variables: Applied Label Encoding to convert categorical features to numerical format
- Preserved encoders for future predictions on new data

e) Feature Scaling:
- Applied StandardScaler to normalize numerical features
- Ensures all features are on the same scale for model training
- Scaled all features (numerical and encoded categorical) for clustering

f) Data Splitting (Classification):
- 80/20 train-test split with stratification to maintain class distribution
- Random state set to 42 for reproducibility

g) Data Sampling (Segmentation):
- Sampled 20,000 records for clustering to improve computational efficiency
- Maintained random sampling with fixed seed for reproducibility

3. MODEL ARCHITECTURE

==

3.1 Classification Model Architecture

Model Type: Random Forest Classifier

Key Architecture Components:
- Algorithm: Ensemble of decision trees (100 trees)
- Tree Parameters:
  * max_depth: 20 (prevents overfitting)
  * min_samples_split: 10 (minimum samples required to split a node)
  * min_samples_leaf: 5 (minimum samples required in a leaf node)
- Class Balancing: class_weight='balanced' to handle imbalanced dataset
- Parallel Processing: n_jobs=-1 for efficient training
- Random State: 42 for reproducibility

Why Random Forest:
- Handles both numerical and categorical features well
- Provides feature importance rankings
- Robust to outliers and missing values
- Good performance on imbalanced datasets with class weighting
- Less prone to overfitting compared to single decision trees

3.2 Segmentation Model Architecture

Model Type: K-means Clustering

Key Architecture Components:
- Algorithm: K-means clustering with optimal k selection
- Cluster Optimization: Uses silhouette score to determine optimal number of clusters (tested k from 2 to 8)
- Evaluation Metrics:
  * Silhouette Score: Measures how similar objects are to their own cluster vs. other clusters
  * Davies-Bouldin Score: Measures average similarity ratio of each cluster with its most similar cluster
- Initialization: n_init=10 (runs algorithm 10 times with different initializations)

- Random State: 42 for reproducibility

Why K-means:
- Interpretable and easy to understand
- Computationally efficient for large datasets
- Works well with standardized features
- Provides clear cluster assignments for marketing segmentation

4. TRAINING ALGORITHM
==

4.1 Classification Model Training

Training Process:
1. Data Preprocessing: Applied all pre-processing steps described in Section 2
2. Model Initialization: Created Random Forest with specified hyperparameters
3. Training: Fit model on training set (80% of data)
4. Prediction: Generated predictions and probability scores on test set
5. Evaluation: Calculated multiple performance metrics

Hyperparameters Selected:
- n_estimators=100: Balance between performance and computational cost
- max_depth=20: Prevents overfitting while allowing sufficient complexity
- min_samples_split=10: Ensures trees don't split on very small groups
- min_samples_leaf=5: Maintains minimum group size in leaf nodes
- class_weight='balanced': Automatically adjusts weights inversely proportional to class frequencies to handle imbalance

4.2 Segmentation Model Training

Training Process:
1. Data Sampling: Selected 20,000 samples for computational efficiency
2. Data Preprocessing: Applied same preprocessing as classification
3. Optimal Cluster Selection:
   - Tested k values from 2 to 8
   - Calculated silhouette scores and Davies-Bouldin scores for each k
   - Selected k with highest silhouette score
4. Model Training: Trained K-means with optimal k
5. Cluster Analysis: Analyzed characteristics of each cluster
6. Visualization: Created 2D PCA visualization of clusters

Cluster Optimization Algorithm:
- For each k in range(2, max_k+1):
  * Fit K-means with k clusters
  * Calculate silhouette score (higher is better)
  * Calculate Davies-Bouldin score (lower is better)
  * Select k with highest silhouette score

5. EVALUATION PROCEDURE
==

5.1 Classification Model Evaluation

Metrics Used:
1. Accuracy: Overall correctness of predictions (90.7%)
2. Precision: Proportion of positive predictions that are correct (38.4%)
3. Recall: Proportion of actual positives correctly identified (81.0%)
4. F1-Score: Harmonic mean of precision and recall (52.1%)

5. ROC-AUC: Area under the receiver operating characteristic curve (0.947)
6. Confusion Matrix: Detailed breakdown of predictions vs. actuals

Results Summary:
- Accuracy: 90.7% - High overall accuracy
- Precision: 38.4% - Lower precision due to class imbalance (many false positives when predicting high income)
- Recall: 81.0% - Good at identifying high-income individuals
- F1-Score: 52.1% - Balanced metric considering both precision and recall
- ROC-AUC: 0.947 - Excellent discrimination ability

Confusion Matrix Analysis:
- True Negatives (<=$50k predicted correctly): 34,206
- False Positives (>$50k predicted incorrectly): 3,223
- False Negatives (<=$50k predicted incorrectly): 470
- True Positives (>$50k predicted correctly): 2,006

5.2 Segmentation Model Evaluation

Metrics Used:
1. Silhouette Score: Measures cluster quality (higher is better, range -1 to 1)
2. Davies-Bouldin Score: Measures cluster separation (lower is better)
3. Cluster Analysis: Demographic and employment characteristics per segment
4. Income Distribution: Percentage of high-income individuals per segment

Evaluation Process:
- Tested multiple k values (2-8) to find optimal number of clusters
- Selected k with highest silhouette score
- Analyzed key features (age, education, marital status, sex, race, capital gains/losses, weeks worked, wage per hour) for each cluster
- Calculated income distribution within each segment
- Generated marketing insights based on segment characteristics

Segmentation Results:
- Optimal Number of Clusters: 4 segments identified
- Segment 0: 6,518 customers (32.6% of population)
  * Average age: 44.6 years, High school graduates, Married females
  * Average wage: $73.27/hour, 32.4 weeks worked/year
  * High-income percentage: 9.5%
- Segment 1: 5,549 customers (27.7% of population)
  * Average age: 8.7 years, Children, Never married
  * Average wage: $0.15/hour, 0.3 weeks worked/year
  * High-income percentage: 0.0% (children segment)
- Segment 2: 6,228 customers (31.1% of population)
  * Average age: 45.1 years, High school graduates, Married females
  * Average wage: $75.54/hour, 32.4 weeks worked/year
  * High-income percentage: 8.4%
- Segment 3: 1,705 customers (8.5% of population)
  * Average age: 39.7 years, High school graduates, Married females
  * Average wage: $49.26/hour, 26.2 weeks worked/year
  * High-income percentage: 4.5%

6. INTERESTING FINDINGS AND EXPLORATION
==

6.1 Feature Importance Insights

Top 10 Most Important Features for Income Prediction:
1. Detailed Occupation Recode (12.5% importance)

2. Weeks Worked in Year (11.7% importance)
3. Age (10.3% importance)
4. Dividends from Stocks (7.4% importance)
5. Number of Persons Worked for Employer (7.4% importance)
6. Detailed Industry Recode (6.6% importance)
7. Sex (4.9% importance)
8. Education (4.7% importance)
9. Detailed Household and Family Stat (3.9% importance)
10. Capital Gains (3.7% importance)

Key Observations:
- Employment-related features (occupation, weeks worked, employer size, industry) collectively account for over 30% of predictive power, making them the strongest indicators of income level
- Age is a significant factor (10.3%), likely correlating with career progression, experience, and accumulated wealth
- Investment income features (dividends, capital gains) together account for over 11% importance, indicating that investment behavior is a strong predictor of high income
- Education and marital status have moderate importance (4.7% and 2.9% respectively), suggesting they contribute but are not the primary drivers
- Sex shows importance (4.9%), potentially indicating gender-based income disparities in the dataset
- Household characteristics (family status, household summary) provide additional context for income prediction

6.2 Class Imbalance Challenge

- The dataset is highly imbalanced (93.8% <=$50k vs. 6.2% >$50k)
- This imbalance affects model performance, particularly precision
- The model correctly identifies 81% of high-income individuals (good recall)
- However, when the model predicts high income, only 38.4% are correct (lower precision)
- This is a common trade-off in imbalanced classification problems

6.3 Model Performance Insights

- The high ROC-AUC score (0.947) indicates excellent discrimination ability
- The model can effectively rank individuals by income probability
- Lower precision indicates the model predicts high-income more liberally, resulting in false positives
- The balanced class weights help the model learn from the minority class

6.4 Segmentation Findings

- The optimal number of clusters was determined through systematic evaluation
- Four distinct customer segments were identified from the 20,000 sample
- Key Segmentation Insights:
  * Segment 1 represents children (average age 8.7 years), comprising 27.7% of the population - this is a unique demographic requiring special marketing consideration
  * Segments 0 and 2 are similar in demographics (married females, high school graduates, mid-40s) but differ in employment patterns and represent the largest adult segments (32.6% and 31.1% respectively)
  * Segment 3 is the smallest (8.5%) with lower wage rates and fewer weeks worked, representing a distinct lower-employment segment
  * All segments show low high-income percentages (0-9.5%), indicating the sampled population is predominantly lower to middle income
- Segments can be used for targeted marketing strategies:
  * Segment 1 (Children): Family-oriented products, educational content,

  parental decision-making focus
  * Segments 0 & 2 (Working Adults): Value-based products, practical solutions, family-focused messaging
  * Segment 3 (Lower Employment): Budget-friendly products, affordability messaging, flexible payment options

7. BUSINESS JUDGMENT AND DECISIONS
==

7.1 Data Approach Decisions

Decision 1: Handling Missing Values
- Rationale: Missing values ('?' and NaN) were common in the dataset
- Approach: Used median imputation for numerical features and 'Unknown' category for categorical features
- Business Impact: Preserves all data points, maximizing usable information
- Alternative Considered: Dropping rows with missing values (rejected due to significant data loss)

Decision 2: Feature Selection
- Rationale: 'weight' and 'year' columns are not predictive features
- Approach: Removed these columns from feature set
- Business Impact: Cleaner model, faster training, better interpretability
- Alternative Considered: Including all columns (rejected to avoid noise)

Decision 3: Data Sampling for Segmentation
- Rationale: Full dataset (199,523 rows) is computationally expensive for clustering
- Approach: Sampled 20,000 records for clustering (approximately 10% of data)
- Business Impact: Faster processing while maintaining representative segments. The sample size of 20,000 is statistically significant and provides reliable segmentation patterns that can be applied to the full population.
- Alternative Considered: Using full dataset (rejected due to computational constraints and diminishing returns on cluster quality)

7.2 Model Selection Decisions

Decision 1: Random Forest for Classification
- Rationale:
  * Handles mixed data types (numerical and categorical)
  * Provides feature importance for business insights
  * Robust to outliers and missing values
  * Good performance on imbalanced data with class weighting
- Business Impact: High accuracy (90.7%) and excellent discrimination (ROC-AUC 0.947)
- Alternatives Considered:
  * Logistic Regression: Simpler but lower performance on complex patterns
  * Gradient Boosting: Similar performance but less interpretable
  * Neural Networks: Higher complexity without significant performance gain

Decision 2: K-means for Segmentation
- Rationale:
  * Interpretable and easy to explain to business stakeholders
  * Computationally efficient
  * Works well with standardized features
  * Provides clear cluster assignments
- Business Impact: Actionable customer segments for marketing
- Alternatives Considered:
  * DBSCAN: More complex, harder to interpret, requires parameter tuning

  * Hierarchical Clustering: Computationally expensive for large datasets

Decision 3: Class Balancing Strategy
- Rationale: Dataset is highly imbalanced (93.8% vs. 6.2%)
- Approach: Used class_weight='balanced' in Random Forest
- Business Impact: Model can identify high-income individuals (81% recall) despite class imbalance
- Alternatives Considered:
  * SMOTE (oversampling): More complex, can introduce noise
  * Undersampling: Loses valuable data
  * Cost-sensitive learning: Requires domain expertise to set costs

7.3 Model Usage Recommendations

Recommendation 1: Classification Model Usage
- Primary Use Case: Identify high-income prospects for premium product targeting
- Recommended Approach: Use probability scores (predict_proba) rather than binary predictions
  * Set threshold based on business needs (precision vs. recall trade-off)
  * For high-precision needs: Use higher threshold (fewer false positives)
  * For high-recall needs: Use lower threshold (catch more high-income individuals)
- Business Value: Enables targeted marketing to high-value customers

Recommendation 2: Segmentation Model Usage
- Primary Use Case: Create personalized marketing campaigns for different customer segments
- Recommended Approach:
  * Use cluster characteristics to design segment-specific messaging
  * Allocate marketing budget based on segment size and income potential
  * Develop product recommendations per segment
  * Test and refine campaigns based on segment response rates
- Specific Segment Strategies:
  * Segment 1 (Children - 27.7%): Target parents with family products, educational offerings, and child-focused services. This segment requires indirect marketing through parents/guardians.
  * Segments 0 & 2 (Working Adults - 63.7% combined): Focus on value-based products, family solutions, and practical offerings. These segments represent the largest market opportunity.
  * Segment 3 (Lower Employment - 8.5%): Emphasize affordability, flexible payment plans, and budget-conscious options.
- Business Value: Improves marketing ROI through personalization and enables efficient resource allocation across distinct customer groups

Recommendation 3: Model Maintenance
- Regular Updates: Retrain models periodically as new data becomes available
- Performance Monitoring: Track model performance metrics over time
- Feature Monitoring: Monitor feature distributions for data drift
- Business Feedback: Incorporate business feedback to refine models

Recommendation 4: Integration Strategy
- Real-time Predictions: Deploy classification model for real-time prospect scoring
- Batch Segmentation: Run segmentation model periodically to update customer segments
- Dashboard Creation: Build dashboards showing model predictions and segment insights
- A/B Testing: Test different marketing strategies per segment

7.4 Limitations and Future Improvements

Current Limitations:

- Classification Model: Lower precision (38.4%) due to class imbalance, resulting in false positives when predicting high income. This may require threshold tuning based on business cost-benefit analysis.
- Segmentation Model: Based on 20,000 sample (10% of data) for computational efficiency. While statistically significant, full dataset clustering could reveal additional nuances.
- Data Recency: Census data may not reflect current economic conditions or recent demographic shifts.
- Feature Engineering: Limited feature engineering beyond basic encoding and scaling. Domain-specific features could improve performance.

Future Improvement Opportunities:

- Advanced Feature Engineering: Create interaction features, polynomial features, or domain-specific derived features (e.g., income-to-age ratio, employment stability metrics)
- Ensemble Methods: Combine multiple models (Random Forest, Gradient Boosting, Neural Networks) for improved classification performance
- Hyperparameter Optimization: Use grid search or Bayesian optimization to fine-tune model parameters
- Alternative Clustering: Explore hierarchical clustering or DBSCAN for potentially better segmentation
- Real-time Model Updates: Implement online learning or periodic retraining with new data
- Explainability: Add SHAP values or LIME explanations for model interpretability
- Cost-Sensitive Learning: Incorporate business costs (false positive vs. false negative) into model optimization

8. REFERENCES

==

1. Scikit-learn Documentation
   - Random Forest Classifier: https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html
   - K-means Clustering: https://scikit-learn.org/stable/modules/generated/sklearn.cluster.KMeans.html
   - StandardScaler: https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.StandardScaler.html

2. Pandas Documentation
   - Data manipulation and analysis: https://pandas.pydata.org/docs/

3. NumPy Documentation
   - Numerical computing: https://numpy.org/doc/

4. Machine Learning Best Practices
   - Handling imbalanced datasets: Chawla, N. V., et al. "SMOTE: synthetic minority over-sampling technique." Journal of artificial intelligence research 16 (2002): 321-357.

5. Clustering Evaluation Metrics
   - Rousseeuw, P. J. "Silhouettes: a graphical aid to the interpretation and validation of cluster analysis." Journal of computational and applied mathematics 20 (1987): 53-65.
   - Davies, D. L., & Bouldin, D. W. "A cluster separation measure." IEEE transactions on pattern analysis and machine intelligence 2 (1979): 224-227.

6. Feature Engineering
   - Guyon, I., & Elisseeff, A. "An introduction to variable and feature selection." Journal of machine learning research 3.Mar (2003): 1157-1182.

7. Random Forest Algorithm
   - Breiman, L. "Random forests." Machine learning 45.1 (2001): 5-32.

8. K-means Clustering
   - MacQueen, J. "Some methods for classification and analysis of multivariate observations." Proceedings of the fifth Berkeley symposium on mathematical statistics and probability. Vol. 1. No. 14. 1967.