Income Classification and Customer Segmentation — Project Report

Author: Sree Harsha Koyi
Date: 2025■09■04

Income Classification and Customer Segmentation — Project Report

Author: Sree Harsha Koyi Date: 2025■09■04

<!-- Notes for reviewers: - I focused on clear, reproducible steps and practical recommendations. - I kept plots and metrics grounded in the generated artifacts under reports/. - Where reasonable, I added plain-language commentary on tradeoffs and next steps. -->

1) Objective

- Build a classifier to predict whether an individual's income is below $50k or at/above $50k using weighted CPS data (1994–1995).
- Build a segmentation model to group people into actionable marketing personas and outline how to target them.

2) Data Understanding

- Source: Weighted census data (CPS 1994–1995). Each record has a `weight` (population representativeness) and `label` (income < or ≥ $50k).
- Files: `census-bureau.data` (values), `census-bureau.columns` (headers).
- Features: 40 demographic/employment fields + `weight` + `label`.
- Missing values: encoded as `?` in categoricals. Some numeric-like columns are strings with non-numeric tokens (e.g., "Not in universe").
- Target mapping: any `label` that contains a `+` is mapped to 1 (≥ $50k). Others map to 0 (< $50k).

3) Preprocessing & Feature Engineering

- Numeric columns: coerced to numeric (invalid tokens -> NaN), median imputation, standard scaling.
- Categorical columns: `?` -> NA, most-frequent imputation, one-hot encoding (ignore unseen categories at inference).
- We excluded `weight` from features and used it as `sample_weight` during training and metric calculation (to reflect the population).
- Before modeling, we applied feature selection and feature extraction on the preprocessed matrix to balance signal and simplicity (no tuning):
- Quick note: One column (`family members under 18`) had all invalid/empty numeric values in some folds and was skipped by median imputation — this is fine, it contributes no numeric signal in those folds.

4) Classification Modeling

- Approach: prioritize clarity over tuning. We use three fixed classifiers with thoughtful defaults and compare them under three fixed feature stages:
- Model selection: 5-fold weighted ROC AUC on the training split; pick the best.
- Final evaluation: held-out test split with weighted metrics.

Results - See `reports/classification_metrics.json` for the selected best model, cross-validated ROC AUC on training, and weighted test metrics. The file also includes a small `candidates` list summarizing pipelines and their CV scores (for transparency, not tuning).

5) Thought Process & Approach - Keep the solution reproducible and communicable: embed preprocessing in the pipeline, choose a simple and interpretable baseline (LogReg), and add two widely used nonlinear baselines (RF, XGBoost) to capture interactions. - Favor dimensionality control for one-hot features (MI selection, SVD) to reduce variance and speed up training, without searching over many options. - Use weighted ROC AUC for selection because it is threshold■independent and respects the survey weights. - Hold out a test split; select models using CV on the training data only to avoid leakage.

6) Questions & Assumptions - Label definition: does "≥ $50k" reflect the current business threshold, or should we adapt to inflation/region/category? - Business objective: should we optimize recall (reach more high■income prospects) or precision (avoid wasted spend)? What is the approximate cost per false positive/negative? - Fairness constraints: are there protected attributes or proxies we must monitor (e.g., race, sex, citizenship)? Any constraints on using certain variables for targeting? - Timeliness: how often will the data refresh? Do we need a retraining cadence and drift monitoring? - Success metrics: besides classification metrics, what downstream KPIs matter (conversion rate, AOV, LTV)?

7) Suggestions & Future Work - EDA: add summary statistics and distributions for key features (weighted), explore missingness patterns, and check target rate stability across cohorts. - Threshold strategy: calibrate probabilities (Platt/Isotonic), then pick operating points by business costs; provide a threshold–metrics curve. - Fairness: run subgroup metrics, calibration checks, and if necessary post■processing (equalized odds tradeoffs) per policy. - Explainability: report permutation importance and SHAP summaries for the chosen model; validate stability

across CV folds. - Robustness: add simple drift monitors (population stability index or KL divergence) and re■train triggers. - Segmentation: iterate on k selection with business input; consider soft clustering (GMM) or hierarchical methods for better persona narratives. - Data pipeline: formalize schema checks, versioned datasets, and model cards for governance.

Comments - The ROC AUC is strong, indicating good ranking power. Accuracy is high given class balance and weights. - Recall (~0.40) suggests the default 0.5 threshold favors precision. For broader outreach (catch more ≥$50k), we can lower the threshold (e.g., 0.35–0.45) to trade some precision for recall. For high-cost campaigns, keep a higher threshold to preserve precision.

Plots - ROC Curve: see `../reports/plots/roc_curve.png` - Confusion Matrix (weighted counts): see `../reports/plots/confusion_matrix.png` - Threshold–metrics curve (precision/recall/F1/accuracy vs threshold): see `../reports/plots/threshold_metrics.png` and table `../reports/threshold_metrics.csv`. - Global permutation importance (directional): see `../reports/permutation_importance.csv`.

5) Segmentation Modeling

- Approach: KMeans on the same preprocessed feature space (one-hot + scaled numerics).
- Chosen k: 6 to balance interpretability and variation across groups.
- Silhouette score (k=6): 0.192 — acceptable for high-dimensional, mixed-type data after one-hot.
- Deliverables: record-level assignments (`outputs/segments.csv`) and weighted profiles (`reports/segment_profiles.csv`).

Segment Highlights (weighted summaries) - Cluster 1 — Children/Dependents - Age ~ 8.6; “Not in universe” for work/industry; 99.9% never married. - This segment represents dependents rather than income earners. For marketing, they’re relevant only via household targeting (e.g., family products, back-to-school), not income-based segmentation. - Cluster 3 — Older Adults, Mostly Not in Labor Force - Age ~ 59.4; majority female; employment fields largely “Not in universe”. - Recommend: healthcare services, retirement planning, leisure activities with senior discounts, delivery convenience. - Cluster 2 — High-Earning Professionals - Age ~ 47.2; high capital gains and dividends; top education “Bachelor’s”; top occupation “Professional specialty”; male-skewed. - Recommend: premium/affluent products, financial services, travel upgrades, high-end electronics. - Clusters 0 and 4 — Prime Working Age, Retail/Clerical Mix - Age ~ 38; top industry “Retail trade”; top occupation admin support/clerical; married, male-skewed. - Recommend: value-driven offers, convenience retail, loyalty programs, buy-now/pay-later; potential aspirational cross-sell. - Cluster 5 — Full-Time Hourly, Long Weeks - Age ~ 38.7; wage per hour very high average due to outliers; weeks worked ~ 48; male-skewed; manufacturing durable goods. - Recommend: durable goods, automotive, tools/DIY, subscription services with practical value.

Comments - Segments are coherent and actionable: they align with life stage, labor force status, and industry/occupation signals. - For production use, I would tune k (e.g., 4–10), compare silhouette and, more importantly, business interpretability. Consider alternative methods (e.g., Gaussian Mixtures for soft clusters, or hierarchical clustering to reveal substructure).

6) Business Recommendations

Classifier usage - Campaign design: choose a probability threshold aligned with budget and tolerance for false positives. For broad awareness, target top 60–70% by score; for high-CPM channels, target top 20–30%. - Fairness: audit performance by protected attributes (e.g., race, sex, citizenship proxies) to avoid disparate impact in targeting; calibrate probabilities per segment if needed. - Measurement: use the threshold–metrics curve to select operating points; run A/B tests to quantify uplift with control groups, and track conversion, average order value, and CLV.

Segmentation usage - Tie offers to segment needs (examples above). Assign creative/messaging and channels per segment (e.g., paid search for professionals, social for younger households). - Prioritize segments by weighted share and expected ROI. For example, Cluster 0/4 have large weighted share and are ideal for scale campaigns; Cluster 2 is smaller but high value. - Combine with classifier: within each segment, use the classifier score to rank individuals for outreach.

7) Risks & Mitigations

- Sampling bias: weights help, but consider domain shift (today's population vs. 1994–1995). Mitigation: retrain with more recent data; monitor drift.
- Label ambiguity: income "≥$50k" threshold is historical; in practice use real current thresholds or continuous targets for calibration.
- Data leakage: avoided by excluding `weight` from features and using a holdout split; keep an eye on any features that directly encode tax-filer status or similar proxies.
- Stability: one-hot high cardinality can produce sparse tails; we ignore unknowns at inference to avoid runtime failures.

8) Implementation Notes

- Artifacts:
- Reproducibility: set `--seed` in scripts; use pinned dependencies; enable periodic retraining.
- Serving: wrap the `joblib` pipeline in a simple API; log features and predictions for monitoring.

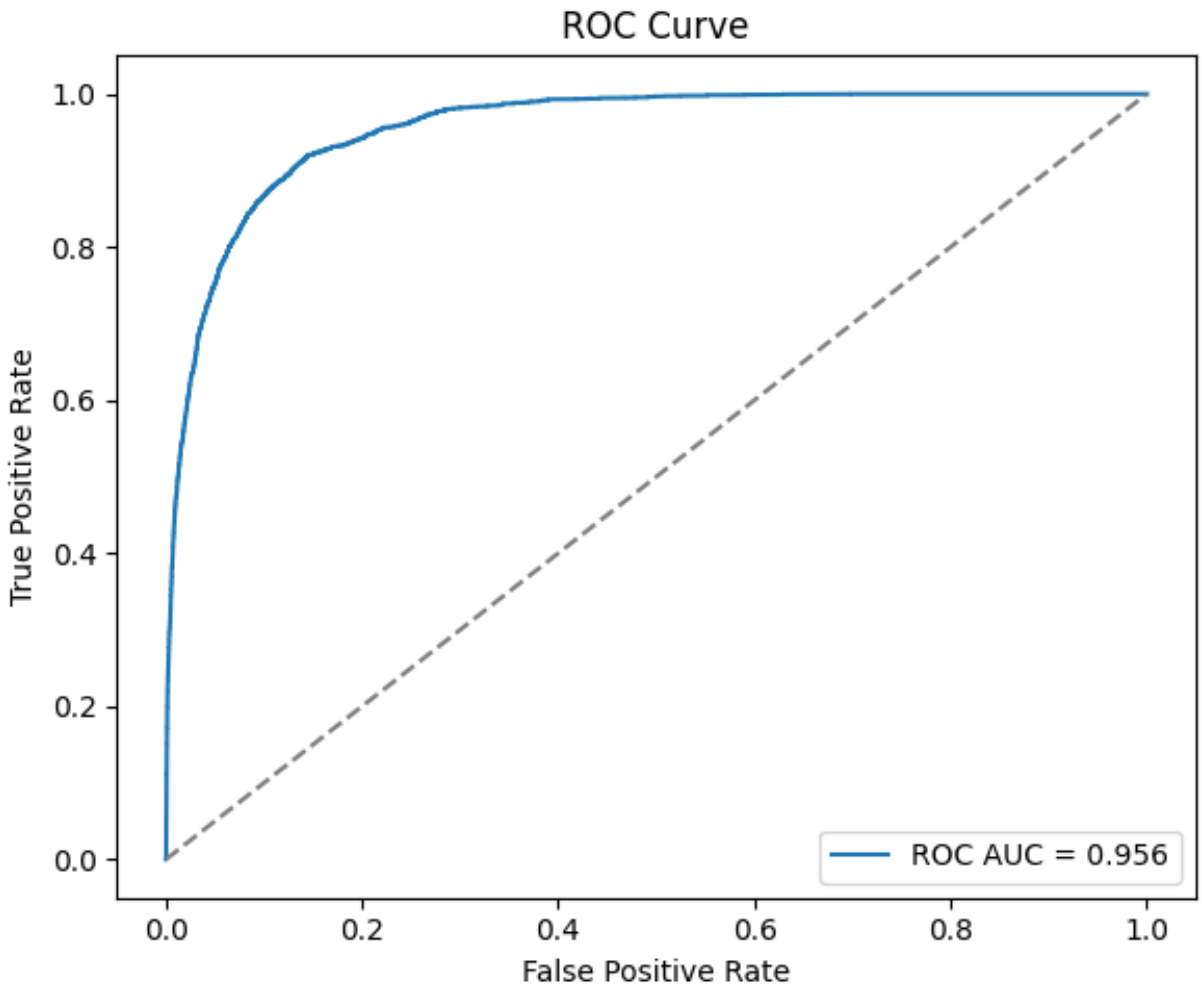
9) References

- scikit■learn User Guide (pipelines, preprocessing, model evaluation)
- Hastie, Tibshirani, Friedman — The Elements of Statistical Learning (classification & clustering fundamentals)
- Kaufman & Rousseeuw — Finding Groups in Data (silhouette and clustering diagnostics)

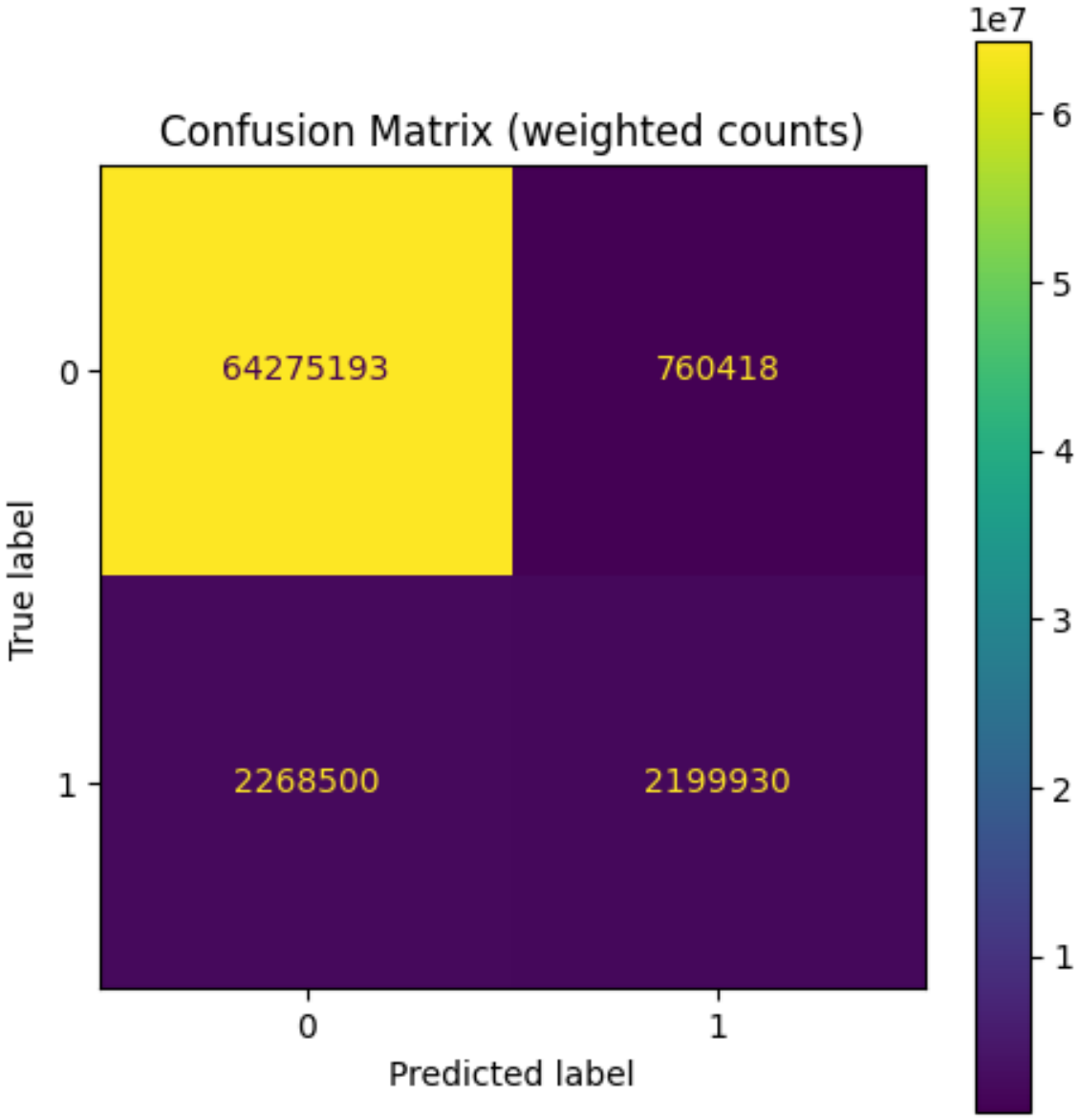
<!-- End of report -->

Figures

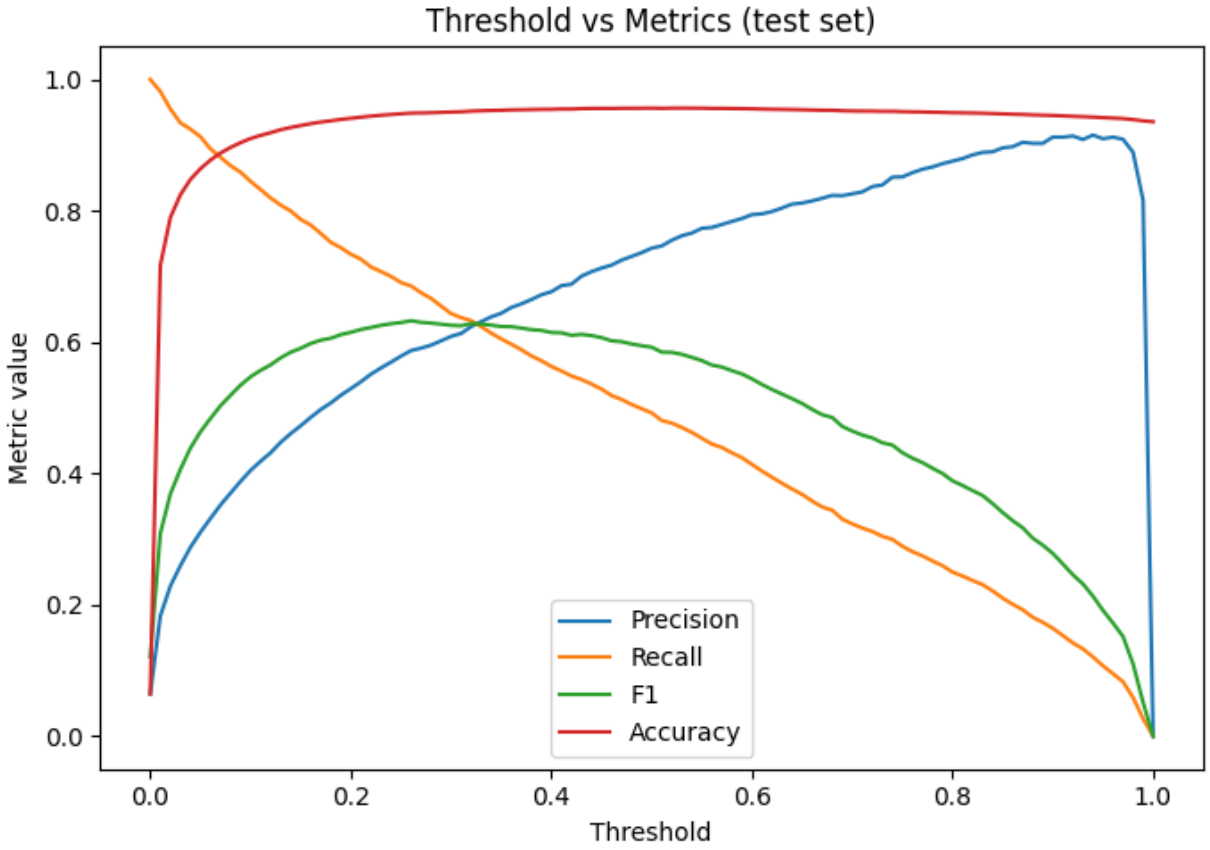
roc_curve.png



confusion_matrix.png



threshold_metrics.png


Threshold vs Metrics (test set)
Metric value
Threshold
Precision
Recall
F1
Accuracy
0.0
0.2
0.4
0.6
0.8
1.0