**Walmart Project Report**

**Project Overview**

**Data:**

* The dataset contains weighted census data from the U.S. Census Bureau’s Current Population Survey (CPS) for 1994 and 1995. It includes 199,523 records and 42 fields: 13 numerical variables (1 float and 12 integers) and 29 categorical variables (e.g., occupation, marital status, education). The dataset is therefore mostly categorical.

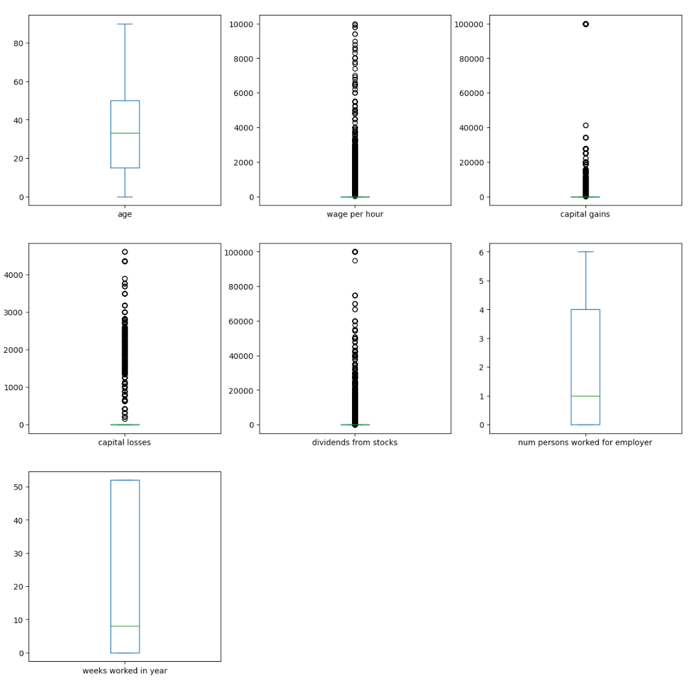
**Objectives:**

* Build a classifier to identify whether an individual earns more than $50,000 or less.
* Develop customer segmentation to inform marketing.

**Data Investigation**

Before using the data, we explored the data to ensure the dataset was valid for building a reliable model

* Profile variables: confirm valid values, spot anomalies, correct data type.
  + We reviewed the distributions of all continuous fields. No material outliers were found for most variables. As expected, wage per hour, capital gains, capital losses, and dividends from stocks show long, right-tailed distributions with extreme yet plausible values. These extremes likely reflect real differences in income and assets, not data errors.



* + Per [U.S. Census Bureau documentation](https://www2.census.gov/programs-surveys/cps/techdocs/cpsmar94.pdf), weight is a survey expansion factor indicating how many people each record represents under stratified sampling. Because this value isn’t available for future model use, like from customers-based data, we’ll use it only for weighting analyses instead of a model feature.
  + "detailed industry recode", "detailed occupation recode", "own business or self employed", "veterans benefits" and "year" need to be changed to “object” type.
  + All categorical variables were checked, and no typos or inconsistencies in category levels were found.
  + The not applicable category appears as “Not in Universe” in some features and '0' in others (but consistent within each feature), which won't impact model performance.
  + Some migration fields (e.g., change in MSA/region, move within region) contain “?”. Other values are valid categories (e.g., “MSA to MSA,” “Same county”).
* Handle missing data: Fill the missing values to “Missing”.
  + Aside from the ‘?’ placeholders (treated as unknowns), the dataset has minimal missing data.”
* Correct data type: Correct data type to the right format
  + "detailed industry recode", "detailed occupation recode", "own business or self employed", "veterans benefits" and "year" need to be changed to “object” type.
* Imbalanced data: check data distribution
  + More than half of the categorical features are dominated by one of the levels, including the label, the data is very imbalanced.

A screenshot of a paper

AI-generated content may be incorrect.

**Objective 1 - Build a classifier to identify whether an individual earns more than $50,000 or less**

Business assumption: For Walmart's customer segmentation strategy, both customer categories are critically important: accurately identifying [high-income customers](https://www.cnbc.com/2025/02/20/how-walmart-won-over-wealthy-shoppers.html) enables premium product marketing, while correctly identifying low-income customers supports value-based product recommendations and pricing strategies.

**Data preprocessing**

In preprocessing,

1. Missing data: Replaced “?” with missing, profiled missingness, and found several categorical columns had >50% missing. Instead of dropping or mode-imputing (which risks information loss/bias), filling missing as “Unknown” to retain signal and keep models usable.
2. Label: Recoded“-50000.”to 0 (≤$50K) and“50000+.”to 1 (>$50K), establishing a clear binary classification target.
3. Weight: The dataset is group-level (each row is a population segment). In production, the true test set will be individual Walmart customers. Since we lack individual-level data now, we train/test on the group-level data and weight it to approximate the future individual-level distribution."
4. Encode: Among the top 10 discrete fields (by unique values), most have >20 categories. The categories have no natural order so we use one-hot encoding, so categories are represented faithfully without implying any sequence.

**Exploratory Data Analysis**

Findings in numerical data:

Correlation analysis shows the strongest positive links to >$50K are weeks worked (~0.26), capital gains (~0.24), number of persons worked for employer (~0.22), and stock dividends (~0.18), with age a smaller signal (~0.14), while survey weight and year are near zero and wage per hour is negligible (~0.02). In practice, prioritize labor- and wealth-related features, keep age as secondary, and de-emphasize near-zero variables.

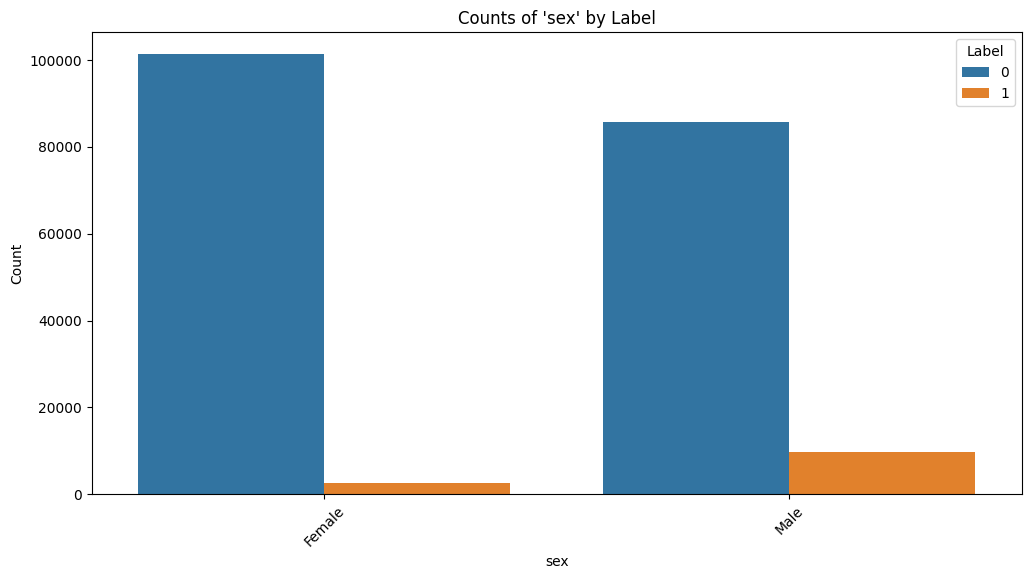
A graph with red squares

AI-generated content may be incorrect.

Double-Click on categorical data:

After conducting a statistical analysis of the relationship between features and income level, it can be clearly observed that there are differences in income distribution among certain features.

Taking gender as an example. The data shows clear income disparities by gender. Among high earners (>$50K), there are 9,719 men compared to 2,663 women, while in the low-income group (≤$50K), women (101,321) outnumber men (85,820). Overall, men are more represented in higher income brackets, while women are concentrated in lower ones, reflecting structural differences in occupation and opportunity during the period.



**Feature Engineering**

After one-hot encoding and combining continuous fields, we reach ~400 features. This high dimensionality likely carries redundant or weak signals that slow training and hurt generalization. To address this, we’ll run feature selection: remove near-constant and highly correlated features, then use tree-based importance on a validation split with early stopping. Early stop ensures we add features only while performance meaningfully improves, preventing overfitting and reducing compute. The final set will keep the most predictive variables, improving accuracy, interpretability, and efficiency.

We ranked features by correlation with the label and trained models using the top N features while tracking F1. Performance improved as N increased and then plateaued; the best trade-off was at N = 31, achieving F1 = 0.41. We use this cutoff to control complexity while preserving accuracy.

**Model Introduction & Metrics**

Three machine learning methods are used here to do the prediction:

1. Logistic Regression: A simple scoring model that adds up key signals and outputs a 0–100% probability; if the score is above a cutoff, it says “yes.” It makes a linear weighted sum of features and runs it through an S-shaped sigmoid to get that probability; also use regularization methods to prevent overfitting.
2. Decision tree: Image a flowchart of short yes or no questions that leads to a decision; the path itself is the explanation. Technically, each split is chosen to maximize purity (e.g., Gini or entropy), and we limit depth or minimum samples per leaf so it doesn’t memorize noise. It handles nonlinear rules well and is easy to show feature importance.
3. Random Forest: Many small decision trees vote on the answer which makes results more accurate and stable. Technically, each tree is trained on a bootstrap sample (bagging) and, at each split, looks at a random subset of features to reduce correlation; we can track out-of-bag (OOB) error as a built-in validation. It’s a strong, robust default, with feature importance available, though less visual than a single tree.

Model Weights: For the model that we implemented a dual weighting strategy to address data imbalance where high-income earners (>$50K) represent only 25% of customers:

1. Class weights prevent the model from ignoring this valuable minority group
2. Sample weights account for stratified census sampling where each record represents different population sizes, using the normalized weights from the original weights.

Metrics: Performance was primarily measured using F1-score for high-income class, balancing precision (avoiding wasted marketing spend on false positives) and recall (capturing actual high earners). This approach maximizes marketing ROI by accurately identifying premium customers while maintaining realistic population-level projections for strategic planning.

Model development: We used a training–validation–test split—training to fit models, validation to tune and select, and a held-out test set for unbiased final evaluation.

**Model Parameter Tuning**

During the model training stage, we divided the data into the training set and the test set in an 8:2 ratio and used the Grid Search method to fine-tune the key hyperparameters of different models to find the best configuration. In the experiment, the above parameter adjustment ranges were set respectively for the three types of models. Through grid search and cross-validation, the optimal parameters for each model are ultimately obtained as follows：

* Logistic Regression：C = 0.1，solver = 'liblinear'
* Random Forest：n\_estimators = 200，max\_depth = 20，min\_samples\_split = 5
* Decision Tree：max\_depth = 30，min\_samples\_split = 2

These optimal parameters can achieve the best performance of the model on the training set, providing a reliable basis for subsequent evaluation on the test set.

**Model evaluation and comparison**

A screenshot of a document

AI-generated content may be incorrect.A graph of a logistic curve

AI-generated content may be incorrect.

Based on the final evaluation, Random Forest was selected as the final model despite Decision Tree's marginally higher F1-score (0.48 vs 0.47) because Random Forest demonstrates significantly superior discriminative capability (AUC 0.93 vs 0.77). All models achieved similar precision challenges (~30-38%) for high-income identification, correctly targeting approximately 2 out of 3 customers in premium marketing campaigns. Random Forest's superior AUC provides greater model reliability and probability estimation accuracy, enabling better threshold optimization for diverse marketing scenarios while maintaining competitive F1 performance for the target demographic.

**Feature importance**

A graph with blue lines

AI-generated content may be incorrect.

After analyzing the Random Forest feature importance, we can clearly identify which variables contribute the most to predicting whether an individual earns above or below $50,000. The top three drivers are detailed occupation recode, weeks worked in year, and age. This indicates that a person’s specific occupation, the consistency of their work history are the strongest determinants of income levelm and their career stage. Following these, features such as num persons worked for employer, and detailed industry recode show that both the type of industry and the size of the employer play significant roles in differentiating income levels. Investment-related variables, including dividends from stocks and capital gains, also emerge as strong signals of higher income. Other influential factors include tax filer status, gender family structure and education level, all of which contribute meaningfully to income classification.

**Insights for Walmart**

These characteristics highlight the main drivers of income disparity and directly inform Walmart’s strategy. Among the top 3 important features, age is straightforward to capture at account creation (date of birth). Weeks worked in a year can be roughly estimated from behavioral patterns, such as shopping frequency, consistency across months, and weekday versus weekend purchasing habits. By contrast, industry and occupation are far harder to observe directly. While purchase history may offer some clues, accuracy is very limited.

To strengthen these signals, Walmart could **consider short, opt-in online surveys** (one or two quick multiple-choice questions) with a small incentive, ensuring responses are collected transparently and with consent.

Moreover, the original dataset is relatively dated and highly imbalanced. Given shifts in technology, e-commerce consolidation, improved delivery, and post-COVID work patterns, we recommend re-examining (and ideally retraining on) recent data to better capture current customer behavior and income signals before applying the model in practice.

**Object 2: Develop customer segmentation to inform marketing.**

**Data preprocessing**

In preprocessing,

1. Missing data: replace all the “?” to nan (we will use one-hot code later to produce a label for it)
2. Categorical data: One hot encode
3. Numerical data: Scaling numeric features to prevent high-magnitude variables from dominating K-means distance calculations.

**Model Introduction & Model tuning**

Kmeans are used in this case. K-means automatically groups data into K clusters by repeatedly assigning each item to its nearest center and updating centers to the group average, making items within a group very similar and items in different groups less similar, which can be a very good methods for us to get customer segmentation.

To tune the K-means model and avoid poor clustering, we will test different numbers of clusters, select the option with the highest silhouette score, and manually check whether the clusters are meaningful for business use.

After investigated the data and a few modeling, when running segmentation, we decided to exclude income and weights but will be included then in the cluster analysis stage. Income would make clusters split largely by income, hiding some real patterns like education, job, and family. Applying weights when training the model would let large groups overpower small but valuable niches. Therefore, we cluster without weights to uncover structure, then profile each segment with weights to reflect true population size, value, and differences.

**Model Result**

To explore potential population structures, we applied KMeans clustering and tested cluster numbers (k = 2–8) using the Silhouette Score. The best score was at k=4 (0.186), but the clusters lacked clear differentiation and were imbalanced in size. At k=5, the score dropped slightly (0.162) but provided a more interpretable and balanced segmentation.

To improve interpretability, we binned working hours and age and applied the dataset’s original weights to better reflect real-world population distributions. Upon review, we found clusters 1 and 4 were very similar, both representing the same group, so we merged them to avoid redundancy. After merging, the resulting clusters present a clearer and more meaningful segmentation, which we use as the basis for further statistical comparison and business interpretation. We should observe more balanced cluster distributions that remain stable when scaled to national population levels.

A screenshot of a computer

AI-generated content may be incorrect.

**Interpret Model Result**

Segment 1: A screenshot of a computer

AI-generated content may be incorrect.

Segment 2

A screenshot of a computer

AI-generated content may be incorrect.

Segment 3

A screenshot of a school age information

AI-generated content may be incorrect.

Segment 4:

A screenshot of a website

AI-generated content may be incorrect.

**References**

1. Adane Nega Tarekegn, Mario Giacobini, Krzysztof Michalak, A review of methods for imbalanced multi-label classification,Pattern Recognition,Volume 118,2021,107965,ISSN 0031-3203,https://doi.org/10.1016/j.patcog.2021.107965.
2. Quinlan, J.R. Induction of decision trees. Mach Learn 1, 81–106 (1986). <https://doi.org/10.1007/BF00116251.>
3. Breiman, L. Random Forests. Machine Learning 45, 5–32 (2001). <https://doi.org/10.1023/A:1010933404324.>
4. K. P. Sinaga and M. -S. Yang, "Unsupervised K-Means Clustering Algorithm," in IEEE Access, vol. 8, pp. 80716-80727, 2020, doi: 10.1109/ACCESS.2020.2988796.