Predicting Income Level Using Census Data: A Comparative Analysis of Machine Learning Models

Amal Hassan
School of Engineering
University of St Thomas
St Paul, Minnesota
hass6113@stthomas.edu

Asia Mohamud
School of Engineering
University of St Thomas
St Paul, Minnesota
moha4323@stthomas.edu

Sadia Mohammed
School of Engineering
University of St Thomas
St Paul, Minnesota
moha6015@stthomas.edu

Abstract—**This report analyzes the U.S. Census Income dataset to predict whether individuals earn above or below $50,000 annually. We use comprehensive data preprocessing, including missing value imputation, feature selection, and class balancing, followed by evaluating five machine learning models. Our analyses identified age, education, and marital status as the strongest predictors of income, while revealing systemic disparities across demographic groups. The Random Forest Classifier achieved the highest performance, demonstrating effectiveness of our methodology.**

Keywords—income prediction, census data, machine learning, classification, feature importance

I. Introduction

Income prediction is a frequently occuring machine learning challenge, where models learn to extract signals from complex patterns. We analyze the 1994 U.S. Census dataset—32,561 records with 14 features related to demographic and employment characteristics—to classify whether individuals earn above or below $50,000 annually. This threshold provides a clean binary split that reflects meaningful economic classification while presenting a realistic 3:1 class imbalance for model testing.

Our work pursues two equally important objectives:

- Technical Performance: Evaluating five classifiers (Logistic Regression, KNN, Decision Trees, Random Forest, and XGBoost) across different feature representations
- Interpretable Insights: Identifying which factors (education, marital status, age) most influence predictions, and how these relationships vary across demographic groups

The analysis reveals which models perform best and why certain patterns persist. For instance:

- Which features (education, age, marital status) drive predictions
- How different algorithms handle inherent class imbalance
- Where models agree or disagree

By combining rigorous modeling with careful interpretation, we transform raw census data into actionable knowledge, demonstrating how our predictive machine learning models help reveal the complex stories behind data.

II. Related Work

Chakrabarty and Biswas (2018) [3] adopted a statistically grounded approach to income prediction. They demonstrated the potential of Gradient Boosting Classifiers for census income prediction, reporting 88.16% accuracy through hyperparameter optimization. While their work provided a valuable case study of ensemble methods on this dataset, they have some key aspects that were not explored:

- Feature selection trade-offs (e.g., retaining fnlwgt despite its low predictive value)
- Demographic disparities in model performance across subgroups
- Handling class imbalance beyond basic preprocessing

Our work extends this foundation by:

- Implementing extensive EDA-driven feature selection (vs. their Extra Trees-only approach)
- Evaluating modern classifiers (XGBoost, Random Forest) not used in their study
- Using oversampling methods to address the class imbalance.

III. Dataset Overview

The dataset used in this project is the UCI Census Income dataset, derived from the 1994 U.S. Census Bureau Current Population Survey (CPS). It was originally collected to support population statistics and government planning. With over 32,000 records, the dataset captures various individual-level demographic and employment characteristics, including age, sex, education, marital status, occupation, and hours worked per week.

While this dataset provides a rich basis for income analysis, it also introduces several data quality and bias concerns. For instance, sampling weights (fnlwgt) were included to ensure representativeness across U.S. states, but this variable was excluded from modeling due to its design-specific role rather than predictive power. Additionally, features such as race and sex may reflect societal inequalities, and thus model outputs must be interpreted with caution to avoid reinforcing existing biases.

The dependent variable in our analysis is binary income classification—specifically, whether an individual earns more or less than $50,000 annually. Predictors include a mix of categorical and numerical features related to education level, work hours, and demographic attributes. This classification problem lends itself well to supervised machine learning techniques, where model performance can be evaluated using accuracy, precision, recall, and AUC metrics.

IV. Exploratory Data Analysis

Before constructing predictive models, we engaged in a thorough exploratory analysis to uncover the hidden stories within the data. This process revealed not only statistical relationships but also systemic patterns that shape economic outcomes. Our journey began with an examination of the dataset's structure and representation, then moved into the three dominant narratives that emerged as pillars of income prediction: education, marital status, and age. Along the way, we discovered how these factors intersect with race and occupation, painting a nuanced picture of the structures in the dataset and socioeconomic realities at large.

A. The Role of Sampling Weights

Every dataset carries a story of how it was collected, and ours was no exception. The *fnlwgt* variable, short for "final weight", served as our starting point. This measure, provided by the Census Bureau, adjusts the survey sample to reflect national population demographics. When we examined median weights across racial groups, a revealing pattern emerged: Black respondents held the highest median weight, indicating they were the most statistically overrepresented group in the dataset compared to their share of the U.S. population. White and "Other" racial categories followed, with Asian and Indigenous groups appearing closer to their true population proportions.

At first glance, this weighting seemed like a technical footnote. But as we progressed, its implications became clearer: while *fnlwgt* was essential for demographic representation and our understanding of the data, we later dropped it as it held little predictive power for our models.

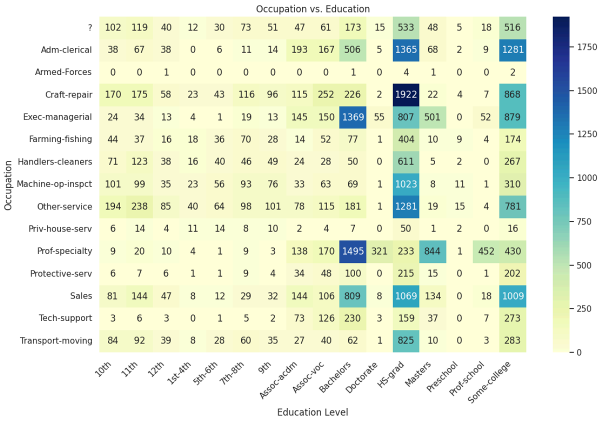
B. Education as an Engine of Mobility

If the data had a headline, it would be this: education pays. Our analysis of the variable *education_num* (a numerical representation of educational attainment) revealed a near-perfect gradient where each additional year of schooling correlated with higher income brackets. The contrast at the extremes was staggering: individuals with advanced degrees (e.g., Master's, Doctorate) were three times more likely to earn over $50,000 annually than those with only a high school diploma. A chi-square test of independence left no doubt about this relationship (χ^2 = 4429.65, *p* < .001).

But numbers alone don't tell the full story. We dug deeper, cross-referencing education with occupation, and uncovered hidden layers: education doesn't just predict income—it *gates access* to the jobs that generate that income. A heatmap of education levels across occupations (Figure 1) showed a stark divide:

- High-Earners: Over 70% of individuals in Executive Managerial and Professional Specialty roles held at least a Bachelor's degree. These occupations accounted for nearly 60% of all high earners (>$50K) in the dataset.
- Manual Labor workers: Conversely, 80% of Handlers-Cleaners and Transport-Moving workers had no education beyond high school. Less than 5% of these workers crossed the $50K threshold.

This wasn't just a correlation; it was a structural reality. The data suggested that education functions as society's sorting mechanism, a lever that lifts some into economic security.



	10th	11th	12th	1st-4th	5th-6th	7th-8th	9th	Assoc-acdm	Assoc-voc	Bachelors	Doctorate	HS-grad	Masters	Preschool	Prof-school	Some-college
?	102	119	40	12	30	73	51	47	61	173	15	533	48	5	18	516
Adm-clerical	38	67	38	0	6	11	14	193	167	506	5	1365	68	2	9	1281
Armed-Forces	0	0	1	0	0	0	0	0	0	1	0	4	1	0	0	2
Craft-repair	170	175	58	23	43	116	96	115	252	226	2	1922	22	4	7	868
Exec-managerial	24	34	13	4	1	19	13	145	150	1369	55	807	501	0	52	879
Farming-fishing	44	37	16	18	36	70	28	14	52	77	1	404	10	9	4	174
Handlers-cleaners	71	123	38	16	40	46	49	24	28	50	0	611	5	2	0	267
Machine-op-inspct	101	99	35	23	56	93	76	33	63	69	1	1023	8	11	1	310
Other-service	194	238	85	40	64	98	101	78	115	181	1	1281	19	15	4	781
Priv-house-serv	6	14	4	11	14	8	10	2	4	7	0	50	1	2	0	16
Prof-specialty	9	20	10	4	1	9	3	138	170	1495	321	233	844	1	452	430
Protective-serv	6	7	6	1	1	9	4	34	48	100	0	215	15	0	1	202
Sales	81	144	47	8	12	29	32	144	106	809	8	1069	134	0	18	1009
Tech-support	3	6	3	0	1	5	2	73	126	230	3	159	37	0	7	273
Transport-moving	84	92	39	8	28	60	35	27	40	62	1	825	10	0	3	283

Figure 1: Occupation vs. Education

C. Marital Status Over Occupation

Here, the data delivered its first surprise. When we tested variables head-to-head, *marital status* (married-civ-spouse) emerged as a stronger predictor of high income than occupation itself. Married individuals were 2.8 times more likely to earn over $50K than their never-married counterparts, even when controlling for age. At first, this seemed counterintuitive, how could a personal circumstance outweigh professional standing?

The answer lay in the intersection of age and marital status. When we stratified the data by age group, a clear pattern emerged: the prime earning years (46–55) were dominated by married individuals, who accounted for 61% of high earners in this group. Never-married individuals, meanwhile, clustered in younger age brackets with lower incomes. The data didn't let us infer causation, but the pattern was undeniable: who you marry may matter as much as what you do for work.

D. Age matters–But only up to a point

Income, we discovered, follows a predictable arc, but one with troubling inequities. Plotting earnings by age group (Figure 2), we observed:

- The Ascent: Earnings rose steadily from ages 17–45, with the sharpest jumps occurring in the 30s.
- The Peak: Between 46–55, nearly 40% of workers earned over $50K, the highest rate of any age group.
- The Decline: After 55, incomes dropped steeply, falling to just 7% for those over 65.

But this arc wasn't universal. When we overlaid racial breakdowns, disparities emerged: White and Asian workers consistently out-earned Black and Hispanic workers at every age, even after accounting for education and marital status. For example, while 30% of White workers in their peak earning years (46–55) crossed the $50K threshold, only 15% of Black workers did. The data hinted at systemic barriers that persist across generations.

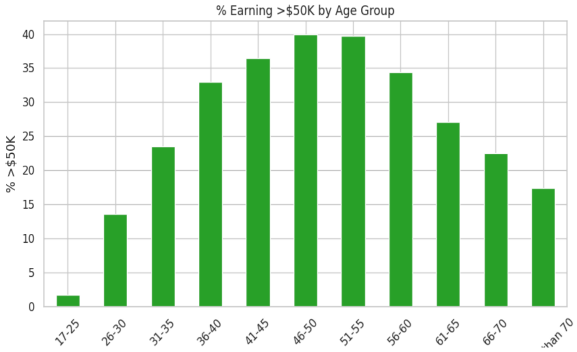


Figure 2: % Earning >$50K by Age Group

Exploratory analysis revealed how income patterns emerge from measurable demographic forces. Three variables, education, marital status, and age, proved to be dominant predictors, each demonstrating consistent, quantifiable relationships with earnings. Education showed near-linear correlation with income brackets, marital status unexpectedly outweighed occupational factors, and age followed a predictable trajectory with notable outliers. These patterns informed our modeling approach by highlighting which variables carried predictive power while surfacing relationships that required further investigation. The analysis provided both a framework for building accurate classifiers and a diagnostic tool for understanding the dataset's underlying structure.

V. Preprocessing

To set the stage for our machine learning models, we needed to transform the dataset into a dataset with clean values, balanced classes, and standardized scales. To achieve this, we went through several preprocessing steps to prepare our dataset while preserving meaningful patterns:

A. Handling Missing Values

All instances of the placeholder '?' were converted to NaN values to properly represent missing data. Rather than handling missing values immediately, we deferred imputation to the pipeline stage where these operations could be properly contained within our cross-validation workflow. This approach prevented any potential data leakage between training and test sets.

B. Feature Selection and Dropping

We systematically evaluated each feature's predictive value and redundancy:

- The education column was removed as it duplicated the ordinal information in education_num
- Capital_gain and capital_loss features were excluded due to extreme zero-inflation (99% zeros)
- Native_country and occupation columns were dropped due to high missingness rates and sparse categories
- Fnlwgt was removed as it served no predictive purpose

C. Addressing Class Imbalance

The original dataset showed significant imbalance, with 76% of instances in the ≤50K class. We applied random oversampling to balance the classes, which was particularly impactful for male respondents.

To visually demonstrate the effects of class balancing, we included pie charts showing income distribution before and after oversampling, specifically for the male subgroup. Before oversampling, only 30.6% of males earned >50K (Figures 3-4). After resampling, this proportion increased to 58.1%, resulting in a more balanced distribution for training. While this artificial balancing risks overfitting to specific examples, it was necessary to prevent models from simply predicting the majority class.

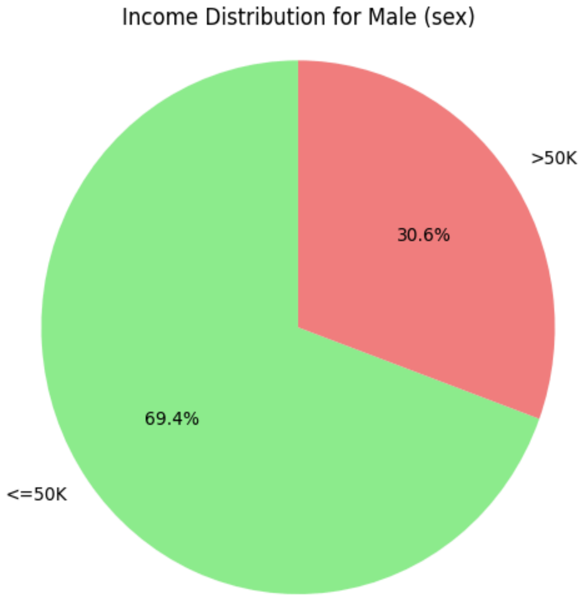


Figure 3: Men Income Distribution-Before Sampling

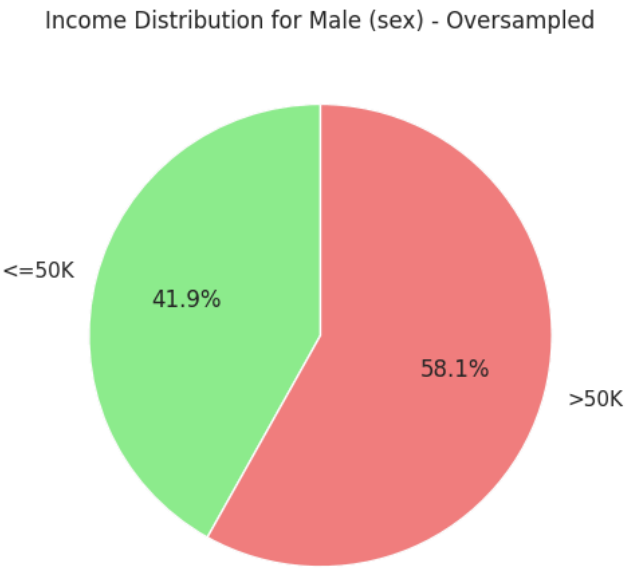


Figure 4 : Men Income Distribution-After Sampling

D. *Pipeline Construction*

Preprocessing demands consistency. Our ColumnTransformer pipeline was to ensure consistent transformation during model training and evaluation:

- Numerical features: Standardized after median imputation
- Categorical features: One-hot encoded after most-frequent imputation
- Target variable: Converted to binary labels (≤50K=0, >50K=1)

The pipeline was fitted exclusively on the training data (80% split) and then applied to the test set, with sparse matrices converted to dense arrays where required by specific algorithms.

This rigorous preprocessing regimen didn't just clean the data, it fundamentally transformed it into optimized data for machine learning. The choices made here, what gaps to fill, which features to remove, which imbalances to correct, would go on to impact the model's performance.

VI. Methodology

A. *Feature Selection:*

Backward Elimination via statsmodels.OLS (α=0.05) was selected to prune irrelevant features systematically. These reduced our one-hot encoded categorical variables to 23 statistically significant predictors. Key features that were retained included age, education_num, hours_per_week and several one-hot encoded categorical variables such as race_White and sex_Male. Variables like workclass_Private and marital_status_Divorced were eliminated. This feature selection technique balanced the simplicity of the model by removing multicollinearity features while preserving interpretability

B. *Dimensionality Reduction:*

We investigated both linear and non-linear dimensionality reduction techniques to improve computing efficiency and evaluate model separability. We focused on all dimensionality reduction techniques that were suitable for classification:

- Linear Discriminant Analysis (LDA): Generated a single discriminative component for binary classification, used for both visualization and as input to linear classifiers.
- Principal Component Analysis (PCA): We reduced the feature space to two principal components to visualize class separability and assess variance retention.
- Kernel PCA (RBF): We further employed RBF-based kernel PCA to capture nonlinear structures in the data for comparative evaluation

C. *Classification Models*

Each classifier was trained on five distinct feature transformations: the Full Feature set, the backward elimination-reduced set, and transformed data derived from linear discriminant analysis (LDA), Principal Component Analysis (PCA), and Kernel PCA.

Each model used in this project was selected based on their strengths in classification problems. We used five classifiers using various configurations.

- Logistic Regression: A strong linear baseline known for its interpretability and efficiency. Suitable for scenarios where the decision boundary is approximately linear.
- K-Nearest Neighbors (KNN): A non-parametric model ideal for capturing local structures in data. Particularly useful when decision boundaries are non-linear and well-separated in feature space.
- Decision Tree: A simple, interpretable model capable of learning non-linear relationships and interactions among features. Effective in handling categorical and numerical data without preprocessing.
- Random Forest: A robust ensemble model that mitigates overfitting by averaging multiple decision trees. Well-suited for high-dimensional, noisy datasets.
- XGBoost: A high-performance gradient boosting method known for its predictive accuracy and handling of class imbalance. Effective in capturing complex feature interactions

C . *Hyperparameter Tuning*

We adjusted each classification model's hyperparameters using GridSearchCV. A parameter grid was established for every model, and internal cross-validation was used in the search procedure to identify the top-performing hyperparameter combinations according to validation accuracy.

- Tree-based Models: For Random Forest and XGBoost, we tuned the number of estimators, maximum tree depth, and other relevant parameters such as minimum samples per split and subsampling rates.

- Distance-based Models: For K-Nearest Neighbors (KNN), we explored different values of k, distance metrics (Manhattan and Euclidean), and weighting schemes (uniform and distance-based).
- Linear Models: For Logistic Regression, we adjusted the regularization strength and solver parameters.

The chosen models were then evaluated on the test set to report final performance metrics i.e., accuracy, precision, recall, and AUC.

VII. Results and Discussion

A. Dimensionality Reduction Impact

We started evaluating our model by examining the performance of dimensionality reduction on the full dataset. Different models were applied to examine performance on all three transformed feature spaces—LDA, PCA, and Kernel PCA, however, Random Forest gave the best accuracy. Specifically:

- LDA (1 component): Achieved 86% accuracy, closely aligning with the full feature model.
- PCA (2 components): Maintained strong performance with 86% accuracy, despite reducing the feature space significantly.
- Kernel PCA (RBF kernel, 2 components): Reached 84% accuracy, indicating that non-linear projections captured meaningful structure, though with a slight drop in predictive power.

These initial results demonstrate that even with dimensionality reduction, Random Forest remained highly effective—highlighting the resilience of ensemble methods and the potential value of compressed feature representations like LDA and Kernel PCA.

B. Reduced Future Model Performance

Backward elimination was used to create a reduced feature set containing only statistically significant predictors. When applied to this streamlined dataset, Random Forest again emerged as the top-performing model, achieving an accuracy of 86%.

Clarification: The classification problem in this study is a binary income, predicting whether an individual earned less than or equal to 50K or more than $50K annually. Therefore, metrics like precision and recall are reported separately for each class ("≤50K" and ">50K"), in the format: precision_minority / precision_majority

Model	Accuracy	Precision	Recall	AUC
Random Forest	0.86	0.93/0.82	0.79/0.94	0.93
KNN	0.85	0.91/0.80	0.77/0.93	0.92
Decision Tree	0.86	0.92/0.81	0.78/0.93	0.89
XGBoost	0.83	0.87/0.79	0.76/0.89	0.91
Logistic Regression	0.80	0.84/0.77	0.74 /0.86	0.88

Table I: Reduced Feature Model Performance

C. Full Feature Model Performance

Model evaluation using the full feature set yielded results closer to those obtained on the reduced dataset. This suggests that the backward elimination process effectively removed redundant or less informative features without sacrificing predictive accuracy. Table II shows that Random Forest remained the strongest performer with 87% accuracy..

Model	Accuracy	Precision	Recall	AUC
Random Forest	0.87	0.93/0.82	0.79/0.94	0.93
Decision Tree	0.86	0.92/0.81	0.78/0.93	0.89
KNN	0.85	0.92/0.80	0.76/0.94	0.92
XGBoost	0.85	0.90/0.81	0.78/0.92	0.92
Logistic Regression	0.80	0.84/0.77	0.74 /0.86	0.88

Table II: Full Feature Model Performance

D. Final Model Selection and Key Insights

Based on the results across all models, the full model using Random Forest was selected as the final model due to its strong and consistent performance. It achieved the highest overall AUC of 0.93 and maintained a high precision and recall across both income classes.

Figure 5 shows the ROC curves for all classifiers trained on the full feature set. The Random Forest curve demonstrates the largest area under the curve AUC)

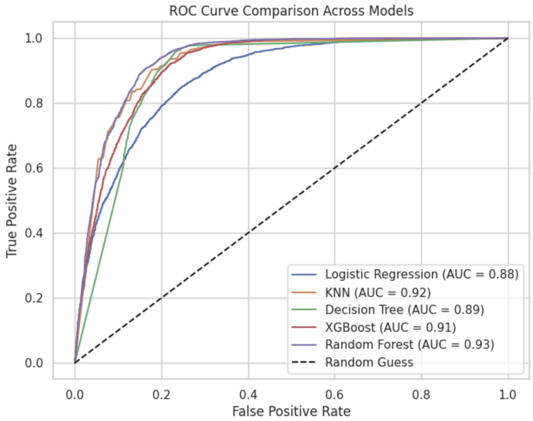


Figure 5: ROC curves comparison Across Models

Figure 6 presents the Random Forest model's top 10 most important features. Features such as age, education_num, and marital_status_Married-civ-spouse played a critical role in income prediction, reflecting both economic and demographic relevance.

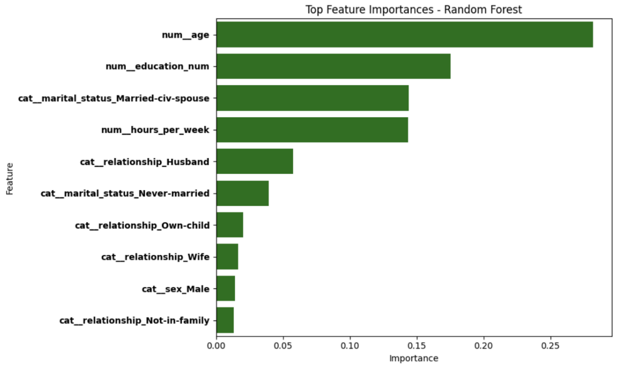


Figure 6: Significant features

Beyond model performance, several key insights emerged:

- Dimensionality reduction (particularly LDA) preserved most of the performance while simplifying the feature space, which may benefit resource-constrained settings.
- Backward elimination effectively reduced noise, achieving results comparable to the full model.
- Class imbalance mitigation using oversampling significantly boosted minority class recall, particularly for KNN and XGBoost.

VIII. Conclusion & Future Work

Following a thorough evaluation of model performance, Random Forest emerged as the most reliable approach for income classification, confirming the strength of ensemble learning in this context. Among the models evaluated, the Random Forest classifier consistently achieved the highest performance, with an accuracy of 87% and an AUC of 0.93 on the full feature set. This success was enabled by a robust and well-preprocessed pipeline, which included class balancing, label encoding, and standardized transformations. While dimensionality reduction and backward feature elimination were explored independently, the full model retained top performance without requiring additional compression or feature pruning.

Dimensionality reduction techniques, including LDA and Kernel PCA, demonstrated that meaningful predictive performance can be retained even with substantial compression of the feature space. In parallel, backward elimination proved effective in simplifying the model and reducing noise, all while maintaining comparable accuracy. Additionally, addressing class imbalance through oversampling played a crucial role in boosting recall for underrepresented income groups, particularly in models such as KNN and XGBoost.

Building on these findings, several promising directions remain for future work:

- Analyze more recent and geographically diverse income datasets to reflect current labor market dynamics and socio-economic trends.
- Explore alternative or more advanced ensemble methods to further enhance prediction robustness.
- Experiment with deep learning architectures, such as neural networks, to capture complex non-linear relationships between features.
- Improve computational scalability for model training and tuning by leveraging faster hardware or more efficient hyperparameter optimization techniques.

Together, these directions aim to refine predictive accuracy while promoting fairness, scalability, and broader applicability in real-world income classification tasks.

IX. References

[1] R. Kohavi and B. Becker, “UCI Adult Census Income Dataset,” UCI Machine Learning Repository, 1996. [Online]. Available: https://archive.ics.uci.edu/dataset/20/census+income

[2] R. Kohavi and B. Becker, “Adult Census Income,” Kaggle, 1996. [Online]. Available: https://www.kaggle.com/datasets/uciml/adult-census-income

[3]N. Chakrabarty and S. Biswas, "A Statistical Approach to Adult Census Income Level Prediction," 2018 International Conference on Advances in Computing, Communication Control and Networking (ICACCCN), Greater Noida, India, 2018, pp. 207-212, doi: 10.1109/ICACCCN.2018.8748528.