CPS 844 Project

Kelvin Chow, Gordon Huang

April 5, 2025

Introduction

This report focuses on applying data mining techniques to predict income levels from census data. Specifically, we use the Adult Income dataset, which contains demographic and employment information collected from the 1994 Census database. The prediction task is to determine whether an individual's annual income exceeds $50,000 based on attributes such as age, education, occupation, and working hours. Income prediction is a significant task that is used in various fields.

This binary classification problem serves as an excellent platform to explore and compare different machine learning algorithms and feature selection techniques. The insights gained can help understand which factors most strongly influence income levels and how different algorithms perform on socioeconomic data.

In this report, we present a comprehensive analysis of the Adult Income dataset, including exploratory data analysis, preprocessing steps, model training and evaluation, and feature selection. We evaluate five different classification algorithms: Logistic Regression, Support Vector Machine (SVM), K-Nearest Neighbours (KNN), Decision Tree, and Random Forest. We also assess how feature selection impacts their performance. Through cross-validation and comparative analysis, we aim to identify the most effective approach for income prediction and gain insights into the key determinants of income levels.

Dataset Description

We used the Adult Income dataset (also known as the “Census Income” dataset). This dataset was based on the census data from 1994. The dataset contains demographic and employment information from the census, with the target variable indicating whether an individual's annual income exceeds $50,000. This makes it a binary classification problem, where we predict one of two classes: ">50K" or "<=50K".

The original dataset contains 32,561 instances with 15 attributes (14 features and 1 target variable). After cleaning the data, there were 24,944 instances. There are both categorical and numerical features. The target is binary classification (income <=50K or >50K).

The dataset includes the following attributes:

- **age**: Continuous numerical feature
- **workclass**: Categorical feature representing employment type (Private, Self-emp, Government, etc.)
- **fnlwgt**: Continuous numerical feature representing the sampling weight
- **education**: Categorical feature representing educational attainment
- **education-num**: Continuous numerical feature representing years of education
- **marital-status**: Categorical feature representing marital status
- **occupation**: Categorical feature representing occupation category
- **relationship**: Categorical feature representing family relationship
- **race**: Categorical feature representing racial background
- **sex**: Binary categorical feature (Male, Female)

- **capital-gain**: Continuous numerical feature representing capital gains
- **capital-loss**: Continuous numerical feature representing capital losses
- **hours-per-week**: Continuous numerical feature representing working hours per week
- **native-country**: Categorical feature representing country of origin
- **income**: Target variable (>50K, <=50K)

The dataset has an imbalanced class distribution:

- Income <=50K: Approximately 74% of instances
- Income >50K: Approximately 26% of instances

This imbalance was important to consider when we were evaluating model performance, as accuracy alone might be misleading. The dataset contained several quality issues that we solved by data preprocessing: missing values denoted by "?" in categorical attributes, high cardinality in categorical features like "native-country" and "occupation", redundant information (e.g., "education" and "education-num" represent similar information), and the "fnlwgt" feature represents census sampling weights and is not directly relevant for prediction.

In order to handle the data processing, we had to drop columns, replace missing values, and drop duplicates. Since the education and education-num columns provide the same information in different forms, we decided to drop the education column since the numerical value of the education column would be more valuable for us. The education-num column was an ordinal numerical column, whereas the education column was a categorical ordinal column. The data in the dataset was separated by commas, followed by a space (“ “). In order to help smooth our data analysis, we removed the spaces separating the data, resulting in data separated only by commas. There were missing values in the dataset denoted by “?”; we decided to drop all rows that had missing values. The columns marital-status, race, and relationship are nominal categorical variables, meaning their values represent distinct categories without hierarchical order. To use them in machine learning models, we applied one-hot encoding, converting each of the categories into a binary column (e.g., race_White, race_Black). This helped prevent algorithms from misinterpreting the categories as numerically ordered values. The sex and income columns were also nominal categorical variables, but the difference was that they only had two values, so we used binary encoding and converted them into values of 0’s and 1’s. For the remaining columns native-country, occupation, workclass they are high-cardinality nominal categorical variables (e.g., native-country has 41 unique values). To avoid the dimensionality curse of one-hot encoding, we applied frequency encoding, replacing each category with its occurrence probability in the dataset.

For algorithms sensitive to feature scale (Logistic Regression, SVM, KNN), we applied standardization. This transforms the features to have zero mean and unit variance, ensuring that features with larger scales don't dominate the learning process. The preprocessed dataset was split into training (80%) and testing (20%) sets using stratified sampling to maintain the same class distribution in both sets.

Model Selection and Training

The classification algorithms we chose were: logistic regression, support vector machine (SVM), k-nearest neighbours (KNN), decision tree, and random forest. To ensure robust evaluation and avoid overfitting, we implemented stratified 5-fold cross-validation. This approach divides the training data into 5 folds while maintaining class distribution, trains the model on 4 folds and evaluates on the remaining fold, repeats this process 5 times, with each fold serving as the validation set once, and then averages the results to obtain a reliable performance estimate.

When we first conducted the cross-validation testing with all features selected, these were the results:

Model Performance (All Features)	
Model	Accuracy
Logistic Regression	0.8337258832372839
SVM	0.8360310699072915
KNN	0.8131295414683037
Decision Tree	0.7917815083938862
Random Forest	0.8381859183162115

For feature selection, we implemented Recursive Feature Elimination (RFE) with Random Forest Classifier. We chose to select the 7 most important features. This works by training the model on all features, ranking features by importance, removing the least important feature, and repeating until the desired number of features is reached. RFE chose the following 7 features:

Top Selected Features	
Feature	Importance
age	0.270448
capital-gain	0.149077
hours-per-week	0.144200
education-num	0.131462
marital-status_Married-civ-spouse	0.127650
occupation	0.108637
relationship_Husband	0.068525

The selected features align well with intuitive expectations about income determinants:

- Demographic factors: Age is the most important feature, suggesting that career progression and experience significantly impact income.
- Education: Education level ranks high, confirming the well-established relationship between educational attainment and earning potential.
- Work-related factors: Hours worked per week, and occupation type are important factors that influence income
- Family status: Being married and being a husband are important predictors, which may reflect both social patterns and historical earning disparities.

Now that we have the 7 most important features of the dataset, we ran the cross-validation testing once again but this time using the 7 most important features and obtained these results:

Model Performance (With Feature Selection)	
Model	Accuracy
Logistic Regression	0.8286143823603107
SVM	0.8352793786018543
KNN	0.8188423953896266
Decision Tree	0.7888248559258331
Random Forest	0.8195439739413681

As we can see, it appears that there were miniscule changes in the accuracy.

Model Performance Difference	
Model	Accuracy Change
Logistic Regression	-0.005111500876973207
SVM	-0.0007516913054371743
KNN	0.0057128539213228136
Decision Tree	-0.0029566524680531003
Random Forest	-0.018641944374843344

With all features, the random forest model performs the best, and with the 7 most important features, SVM performs the best.

After the cross-validation tests were run, we fitted the data to each model and obtained classification reports. The reports contained the metrics:

- Precision: This metric measures how many of the predicted positive instances (income >50K) were actually correct

- Recall: measures how many of the actual positive instances were correctly identified.
- F1-score: the harmonic mean of precision and recall, providing a balance between the two
- Accuracy: the overall percentage of correct predictions made by the model

We obtained the following results:

<table>
<tr><td colspan="4">Logistic Regression</td></tr>
<tr><td></td><td>Precision</td><td>Recall</td><td>F1-score</td></tr>
<tr><td>0 (<=50k)</td><td>0.85</td><td>0.92</td><td>0.89</td></tr>
<tr><td>1 (>50k)</td><td>0.72</td><td>0.55</td><td>0.62</td></tr>
<tr><td colspan="4">Accuracy: 0.83</td></tr>
</table>

Logistic regression performs well overall, but its recall for income >50K is relatively low (55%), meaning it struggles to identify all high-income individuals.

<table>
<tr><td colspan="4">SVM</td></tr>
<tr><td></td><td>Precision</td><td>Recall</td><td>F1-score</td></tr>
<tr><td>0 (<=50k)</td><td>0.86</td><td>0.93</td><td>0.89</td></tr>
<tr><td>1 (>50k)</td><td>0.74</td><td>0.56</td><td>0.64</td></tr>
<tr><td colspan="4">Accuracy: 0.84</td></tr>
</table>

SVM shows slightly better performance than logistic regression, particularly in terms of F1-score (64% for >50K). Its precision and recall scores suggest a good balance.

<table>
<tr><td colspan="4">KNN</td></tr>
<tr><td></td><td>Precision</td><td>Recall</td><td>F1-score</td></tr>
<tr><td>0 (<=50k)</td><td>0.86</td><td>0.89</td><td>0.88</td></tr>
<tr><td>1 (>50k)</td><td>0.76</td><td>0.59</td><td>0.63</td></tr>
<tr><td colspan="4">Accuracy: 0.82</td></tr>
</table>

KNN performs slightly worse than logistic regression and SVM, especially in terms of precision and recall for income >50K.

<table>
<tr><td colspan="4">Decision Tree</td></tr>
<tr><td></td><td>Precision</td><td>Recall</td><td>F1-score</td></tr>
<tr><td>0 (<=50k)</td><td>0.85</td><td>0.88</td><td>0.86</td></tr>
<tr><td>1 (>50k)</td><td>0.61</td><td>0.55</td><td>0.58</td></tr>
<tr><td colspan="4">Accuracy: 0.79</td></tr>
</table>

Decision Tree shows lower accuracy than the other models and performs poorly in classifying income >50K due to lower recall (55%).

<table>
<tr><td colspan="4">Random Forest</td></tr>
<tr><td></td><td>Precision</td><td>Recall</td><td>F1-score</td></tr>
<tr><td>0 (<=50k)</td><td>0.86</td><td>0.89</td><td>0.87</td></tr>
<tr><td>1 (>50k)</td><td>0.62</td><td>0.58</td><td>0.61</td></tr>
<tr><td colspan="4">Accuracy: 0.81</td></tr>
</table>

Random Forest improves upon Decision Tree slightly, offering better precision and recall for income >50K, but still struggles in identifying all high-income individuals.

SVM has the best overall performance, achieving the highest accuracy (84%) and F1-score (64%) for income >50K.

Conclusion

This analysis demonstrates the effectiveness of machine learning in predicting income levels from census data. The combination of careful preprocessing, appropriate algorithm selection, and feature selection yields models with good predictive performance. Moreover, the identified important features align with economic theory, providing validation for both the models and traditional understanding of income determinants.

The balanced approach of comparing multiple algorithms and assessing the impact of feature selection provides valuable insights into the trade-offs involved in model development. While no single approach emerges as universally superior, the results offer guidance for selecting appropriate techniques based on specific requirements for accuracy, efficiency, and interpretability.

In conclusion, this study showcases the potential of data mining for socioeconomic analysis while highlighting areas for future research and improvement.

References

Dua, D., & Graff, C. (2019). *UCI Machine Learning Repository: Adult Data Set*. University of California, Irvine. Retrieved from https://archive.ics.uci.edu/dataset/2/adult