8/22/2025

Advanced Data Mining Final Project Report

Advanced Data Mining for Data-Driven Insights and Predictive Modeling

Sujan Dumaru

Student ID: 005030060

MSCS 634: Advanced Big Data and Data Mining

Dr. Satish Penmatsa

Summer 2025

## Introduction

In this project, I used the UCI Adult Income dataset to practice the complete data mining process. My goals were simple: clean the data, understand the patterns with EDA, build models (regression, classification, clustering), mine association rules, and then give practical recommendations. I also reflect on ethics, because income prediction can be sensitive and can carry unfair bias.

## Dataset and why I chose it

I used the adult dataset from the UCI Machine Learning Repository. It has about 32,000 rows and 15 columns with demographic and work information (age, education, occupation, hours per week, etc.), plus the income label (<=50K or >50K). I chose it because:

* It’s public and well known for machine learning practice.
* It combines numeric and categorical features, so I can show full preprocessing.
* It is relevant to real life (labor and income), which makes insights meaningful.
* It contains missing values, duplicate records, inconsistent categorical formatting, and numeric outliers, making it ideal for data cleaning practice.
* Supports a wide range of preprocessing and EDA techniques, including imputation, standardization, outlier detection, and feature correlation.

## Data preprocessing

**Load and normalize:** I loaded directly from the UCI URL, treated "?" as missing, and used *skipinitialspace=True*. For strings I trimmed spaces and converted them to lowercase for consistency.

**Missing values and duplicates:** I imputed categorical NAs using the mode and dropped exact duplicates.

**Type casting:** I converted numeric columns (age, fnlwgt, education-num, capital-gain, capital-loss, hours-per-week) carefully and filled rare numeric NAs with the median.

**Reasonable ranges and strict filters:** I applied strict IQR removal on capital-gain, capital-loss, and hours-per-week.

**Final shape:** After these steps, my final cleaned DataFrame had 20,302 rows and 15 columns (from the shape logs in the notebook). In an earlier version of my Deliverable 1, I got closer to ~18K rows; the difference comes from slightly different thresholds (for example, how strict IQR and rare-category cutoffs are).

## Exploratory data analysis (EDA) and feature engineering

**EDA highlights:**

* Most people work around 40 hours/week; long-hour workers (>45) are a smaller group.
* Education level and occupation show strong differences across the income label.
* The dataset is imbalanced toward <=50K, which matters for evaluation.

**Feature engineering:**

* I created simple flags like any\_capital\_gain and an hours > 45 flag for rules and clustering.
* Important note: for regression where the target is hours-per-week, I removed any target-derived features like is\_high\_hours to avoid leakage. This made a big difference in honest performance.

## Regression (target: hours-per-week)

**Models:** Linear Regression, Ridge, and Lasso with a preprocessing pipeline (numeric scaling + one-hot for categoricals).

**Leakage fix:** At first, I accidentally included is\_high\_hours (derived from the target). That inflated R² to ~0.74, which is not realistic. After dropping this leakage, results became reasonable.

Results (from the notebook table after the fix):

* Linear Regression: CV R² ≈ 0.103, RMSE ≈ 3.64 hours
* Ridge: CV R² ≈ 0.103
* Lasso: CV R² ≈ 0.104

What this means. Demographics alone do not explain weekly work hours very well. There are many unobserved factors (job contracts, overtime rules, personal choice). Regularization doesn’t help much because the signal is weak. The models are slightly better than the mean baseline, but not strong predictors.

## Classification (target: income >50K vs <=50K)

**Models:** Logistic Regression (baseline), Decision Tree (grid search on depth and leaf size), and KNN (sweep on k). The pipeline used scaling + one-hot encoding and stratified train/test split.

Key metrics from the notebook (Logistic Regression):

* Accuracy: 0.849 on the test set
* ROC-AUC: 0.878
* Class 0 (<=50K): precision 0.884, recall 0.933
* Class 1 (>50K): precision 0.666, recall 0.522

**Interpretation:** Performance is in the normal range for this dataset. The model is better at finding the majority class (<=50K). For >50K, recall is lower, which is common here. Threshold tuning can trade precision vs recall depending on the goal. Decision Tree and KNN were comparable overall; the tree is helpful for interpretability.

## Clustering (unsupervised structure)

**Methods:** K-Means *(k=3)* and Agglomerative *(n\_clusters=3)* on the preprocessed matrix, then PCA to visualize.

**Notebook results:**

* Silhouette (K-Means): ~0.139
* Silhouette (Agglomerative): ~0.093

**Interpretation:** Scores are positive but modest, which is expected for high-dimensional, mixed-type data. Still, clusters show patterns that map to occupation, education, and hours. For example, one cluster tends to include higher-education and longer hours, which overlaps with the high-income class.

## Association rule mining

I built a “basket” from selected categorical columns plus engineered bins like *hours>45* and *capital\_gain>0*. With *min\_support=0.01* and filtering consequents that indicate *income >50K*, the rules table is non-empty in the notebook.

Typical patterns. The top rules often combine:

* Higher education (e.g., bachelor’s, master’s, doctorate)
* Occupation like exec-managerial or professional-specialty
* Working long hours (hours>45)

These antecedents imply income >50K with lift > 1, which means they co-occur more than chance. I did not force exact numbers in the report because support and confidence depend on thresholds, but the qualitative pattern is stable.

## Practical recommendations

1. If the goal is prediction and communication, start with Logistic Regression. It gives strong ROC-AUC and is easy to explain.
2. Focus on education and skills for real interventions. Both clustering and rules show strong alignment of higher education and professional roles with higher income. Training and upskilling make practical sense.
3. Avoid using sensitive attributes in deployment. They can improve short-term metrics but harm fairness and trust. Keep them for auditing only, not for decision logic.
4. If predicting weekly hours is important, collect richer features. Job contract type, overtime policy, and industry-specific variables would help. Current demographic features only give CV R² around 0.10.
5. Tune the classification threshold for the required use cases. If missing high-income cases are costly, raise recall (at the cost of precision) and measure the trade-off with precision-recall curves.

## Ethical considerations

* **Bias and fairness:** The dataset reflects historical inequality (gender, race, occupation). Models can learn these patterns. I reported both overall metrics and class-wise metrics, and I recommend subgroup evaluation (for example, compare precision/recall across groups) before deployment.
* **Sensitive attributes:** Features like sex and race should be excluded from the final decision pipeline to reduce direct discrimination risk. They can be used for fairness analysis only.
* **Privacy and purpose:** Income prediction can affect people’s opportunities. Use models for insight and planning, not to deny services. Keep data secure and anonymized during training.
* **Data cleaning choices matter:** Strict filters (like trimming rare categories) can change who represents the final sample. Be transparent about these choices and test robustness.

## Limitations and future work

* **Limited explanatory power for hours/week:** I need better features to improve regression, such as job level, contract type, or overtime rules.
* **Fairness evaluation:** As next step, I should add subgroup metrics (for example, equalized odds, demographic parity) and test threshold adjustments or re-weighting.
* **Model extensions:** For classification, I can compare gradient boosting (XGBoost/LightGBM) with careful regularization and feature importances. For clustering, I can try HDBSCAN to better handle uneven densities.
* **Association rules coverage:** By lowering support slightly (for example, 0.005), I can see more fine-grained rules, but I will validate them for stability.

## Conclusion

This project covered the full pipeline from raw data to insights. After careful cleaning (final shape ~20K rows), I learned that demographics alone cannot predict weekly hours very well (CV R² ≈ 0.10), but they do separate income groups enough for a good classifier (ROC-AUC ≈ 0.88). Unsupervised clustering and association rules tell a similar story: education level, occupation type, and long work hours relate strongly to higher income. The ethical part is important; we must reduce bias and avoid using sensitive attributes in decisions. Overall, the models are useful for insight and screening, and the recommendations focus on education and fair practice.

**References**

Han, J., Pei, J., & Tong, H. (2022). Data Mining (4th ed.). Elsevier S & T. <https://reader2.yuzu.com/books/9780128117613>

UCI Machine Learning Repository – Adult Dataset.

Wu, D. (2024). Data Mining with Python. Taylor & Francis. <https://reader2.yuzu.com/books/9781040010402>