Wage Prediction Using Machine Learning

Shiva Sai Vummaji

Objective:

This project aims to build a machine learning model to predict individual wages based on human capital, educational qualifications, race & demographics, and occupational features using real labor market data. Beyond prediction (if I am successful and have time), I will assess whether the model amplifies bias along sensitive attributes such as gender and race, connecting the project to themes of wage inequality and fairness in labor markets.

Data and Methodology:

I used the U.S. Census Bureau's Current Population Survey (CPS) or American Community Survey (ACS) microdata, focusing on individuals aged 18–65 in the labor force. The dataset includes relevant variables such as hourly wage, gender, race, age, education, and hours worked per week. For hourly wages, I either got them directly or computed them by dividing annual income by total hours worked annually. I also mapped categorical variables to human-readable labels and then one-hot encoded for modeling. I trained two predictive models:

- Linear Regression (baseline)
- Feedforward Neural Network (PyTorch)

Both models were evaluated on a separate test set using RMSE, MAE, and R^2.

Detailed Steps:

1. Data Collection & Cleaning:
    a. I will get data from IPUMS and select relevant variables, similar to how I did for Data Explorer assignments.
    b. I will convert categorical variables (e.g., SEX, RACE, EDUC) to labels and remove any duplicates/invalid/missing values.
2. Exploratory Analysis:
    a. Understand/graph wage distribution by education, gender, race, etc.
        i. I can even utilize graphs from Data Explorers for this
    b. I will create a correlation matrix for the variables.

c. My goal here would basically be to understand the data better.

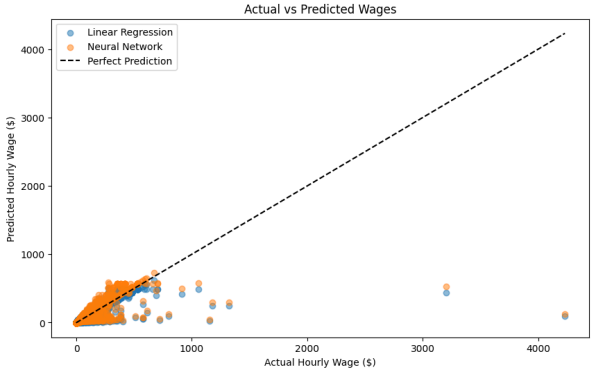
3. Model Building:
    a. I will utilize 2 types of Machine Learning models: Linear/Logistic Regression and Neural Networks.
    b. Linear Regression (sklearn, numpy, pandas):
        i. Split dataset into train/test data
        ii. Using the linear regression library from sklearn, I will build the model and fit it on the training data.
        iii. I will then predict using the test data.
    c. Neural Network (PyTorch, numpy, pandas):
        i. Using the torch module, I will build a neural network and train it on the training data.
        ii. I will then create a testing mechanism.
4. Model Evaluation:
    a. I will utilize metrics such as RMSE, MAE, R^2 to evaluate both Linear Regression and Neural Network.
    b. I will also plot actual vs. predicted wages for both models and compare.

Results:

```
Linear Regression Results:
RMSE: 27.118867589447927
MAE: 4.601646530987878
R2 Score: 0.6332488815307505

Neural Network Evaluation:
RMSE: 27.485950516991423
MAE: 6.969370400999904
R2 Score: 0.6232529448337527
```

Overall, Linear Regression performed slightly better than the Neural Network across all metrics, as the values for RMSE and MAE were lower and R^2 was closer to 1. An R^2 of 0.63 suggests that the Linear model explains about 63% of the variation in wages, which is strong performance for labor market data. However, it is important to note that both models have very similar results for RMSE, so one isn't that much better than another.



I plotted Actual vs. Predicted wages for both models. Most points clustered near the diagonal perfect-prediction line for lower to middle wages ($0–$200/hr). However, both models

struggled more for very high earners ($500+/hr), leading to wider errors, which is common due to the heavy right-tail distribution of income.

Sample Prediction:

For a 30-year old White Male with a Bachelor's Degree working 40 hours per week:

- Linear Regression Predicted Wage: $28.79/hr
- Neural Network Predicted Wage: $26.49/hr
- Actual Average Wage (real): $34.60/hr

The Linear Regression model is closer to the actual average wage, and based on the metrics from before (MAE and RMSE), it is clear that the Linear Regression model is better here.

Interpretation:

1. Linear Regression (Simple Model) was able to outperform overall, as it resulted in lower errors. This can reveal that the relationship between features/predictors and hourly wage might be mostly linear.
2. Neural Network did not outperform Linear Regression, which is probably because the relationship between the features/predictors and hourly wage is not as complex, so we would not require deep learning.
3. People who earned higher wages were harder for the models to predict accurately, since they were mostly outliers and had very high variability.
4. From exploratory data analysis, I also found that the wage distribution between various education groups to be highly variable.

Tech Stack:

1. Programming Language: Python 3.10
2. Data Source: IPUMS
3. Data Analysis: Pandas, Numpy, Scikit-Learn, Matplotlib
4. Machine Learning: PyTorch, Scikit-Learn
5. Environment: Google Colab/Jupyter Notebook
6. Codebase: Github Repository