Income Classification Using Machine Learning

Hemn Sheikholeslami

Introduction

This report details the execution and outcomes of a data mining project aimed at predicting individual income levels based on U.S. census data, using machine learning to classify individuals into income brackets above or below $50,000. The project encapsulates data handling, analysis, and model training within a structured machine learning pipeline.

Data Preprocessing

Comprehensive data preprocessing was conducted to ensure optimal model input quality:

- Missing values marked as '?' were identified and replaced with the mode of their respective columns, ensuring no data point was left unutilized.
- Categorical variables were transformed using one-hot encoding to facilitate numerical analysis.
- Numerical features were standardized to bring them onto a comparable scale, enhancing model sensitivity and accuracy.

Feature Reduction via PCA

The dimensionality of the data was reduced using Principal Component Analysis (PCA), maintaining 95% of the variance while reducing the feature set from 108 to 88. This step significantly reduced computational complexity and improved the handling of multicollinearity.

Model Implementation and Performance

Several machine learning models were evaluated:

- K-Nearest Neighbors (KNN)
- Support Vector Machine (SVM)
- Naive Bayes
- Multilayer Perceptron (MLP)

Performance metrics such as accuracy, precision, recall, and F1-score were calculated for each model, as detailed in the table below:

Performance Metrics for Various Models

Model	Accuracy	Precision ($ $50K)	Recall ($ $50K)	Precision ($>$50K)	Recall ($>$50K)	F1-Score
KNN	82.93%	87%	91%	66%	58%	—
SVM	85.38%	87%	94%	76%	56%	—
Naive Bayes	61.44%	93%	54%	37%	87%	—
MLP	84.00%	88%	92%	69%	59%	—

Visualizations

To further elucidate the findings, several visualizations were utilized:

- Confusion matrices for each model to detail true positives and negatives.
- Precision-recall curves and ROC curves to visualize model performance across different thresholds.

Example of a Confusion Matrix

Example of a Confusion Matrix

Code Cleanliness and Documentation

Code was meticulously documented and cleaned following PEP 8 conventions. The project was developed in Jupyter Notebook, facilitating the iterative experimentation with different algorithms and parameters. This approach not only ensured transparency but also enhanced reproducibility.

Conclusion

This report encapsulates a comprehensive analysis from data preprocessing to detailed model evaluation. The project underscores the effectiveness of machine learning in predictive income classification, with SVM and MLP showing particularly strong performance. Recommendations for future work include further model tuning and exploring alternative dimensionality reduction techniques.