

Glen Morgenstern

DATA 37000

Prof. Edwin Lo

Nov. 11, 2025

Modeling commute time with 1990 US Census data

1. Introduction

This study examines the 1990 United States Census dataset (UC Irvine Machine Learning Repository, ID 116) to predict categorical commute times (dTravtime). The dataset contains over 1.5 million records and 69 discretized variables, transformed by UC Irvine into categorical bins using SQL functions (see Appendix A and the src folder).

The primary objective is to evaluate the predictive performance of classical machine learning algorithms in modeling commute time categories. Multinomial logistic regression, k-nearest neighbors (KNN), random forest, gradient boosting (XGBoost), and support vector machines (SVM) were all used to model commute time. Commute duration is a salient socioeconomic indicator, reflecting labor supply constraints, urban spatial structure, and household decision-making.

2. Data Overview

The dataset consists of 1,534,490 observations with 68 predictors drawn from the 1990 U.S. Census. The target variable, dTravtime, represents commute duration and has been discretized into seven bins ranging from no commute to 60 minutes or more. Predictors span demographic factors such as age and citizenship, occupational measures including industry and occupation codes, socioeconomic indicators such as income and earnings, and household characteristics.

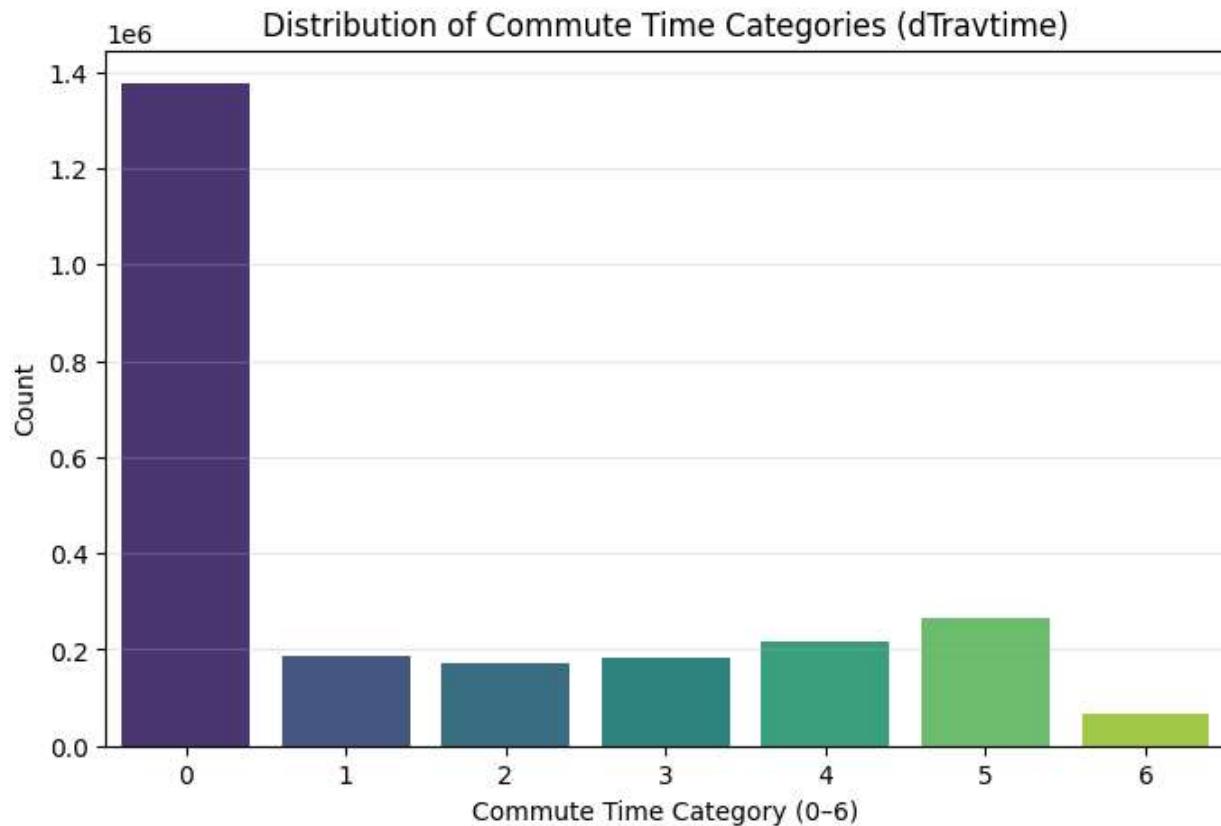
To reduce cardinality and preserve confidentiality, continuous variables were transformed into categorical bins (e.g., discAge, discDepart, discIncome1). While the dataset contained no missing values, certain special codes required interpretation, which is clarified in the SQL appendix documenting the transformations applied by UC Irvine.

3. Exploratory Data Analysis

3.1 Target Distribution

Commute times are highly imbalanced, with short commutes (<30 minutes) dominating the distribution (Figure 1). Longer commutes (≥ 60 minutes, $dTravtime = 6$) are comparatively rare. This imbalance necessitated class balancing via undersampling prior to model fitting.

Figure 1. Distribution of Commute Time Categories (dTravtime)



3.2 Univariate Analysis

Since there are many features included in the data, here are a few features of interest that relate to commute time.

- **Departure time (dDepart):** Represents the time left for work. The majority of individuals depart between 6:00–8:00 AM (Figure 2).
- **Weekly hours (dHours):** Represents working hours per week. We see strong clustering around 40 hours per week (Figure 3).

- **Occupation (dOccup):** A small number of occupational categories dominate the distribution (Figure 4).
- **Industry (dIndustry):** Concentration in manufacturing, services, and trade sectors (Figure 5).

Figure 2. Distribution of Departure Time

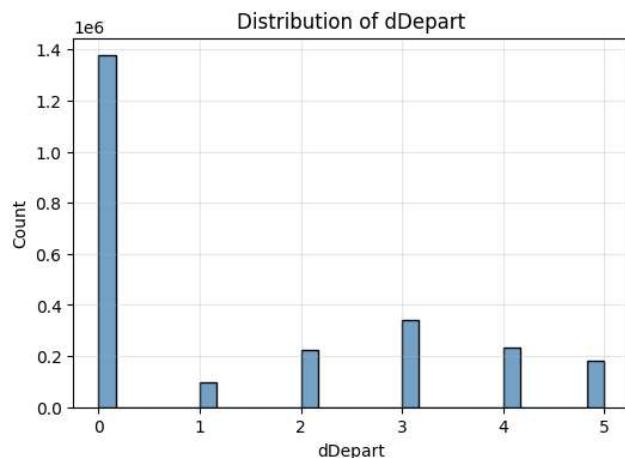


Figure 3. Distribution of Weekly Hours

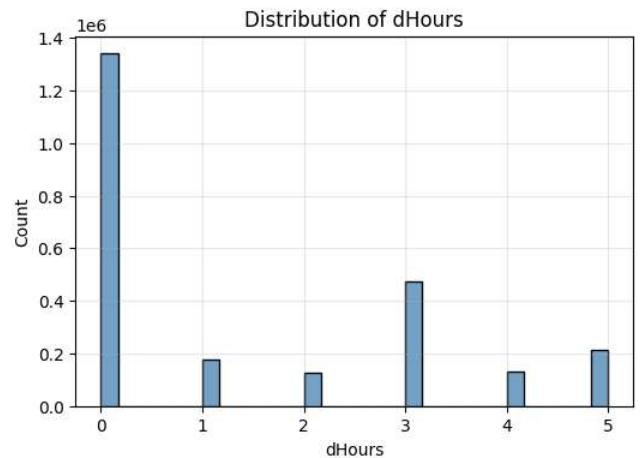


Figure 4. Top 10 Occupation Categories

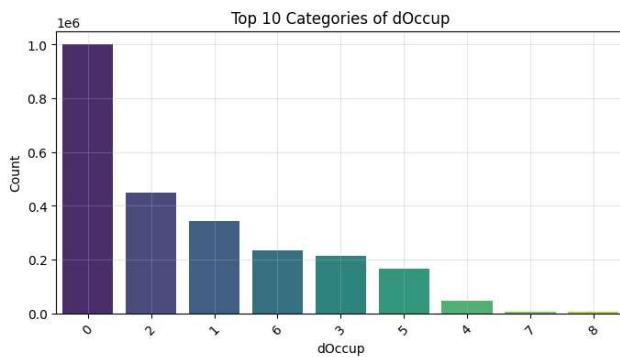
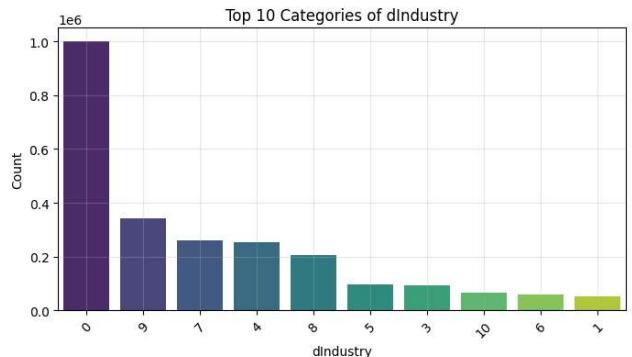
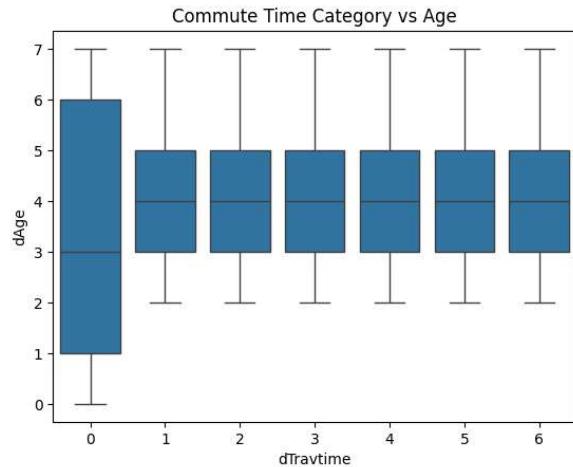
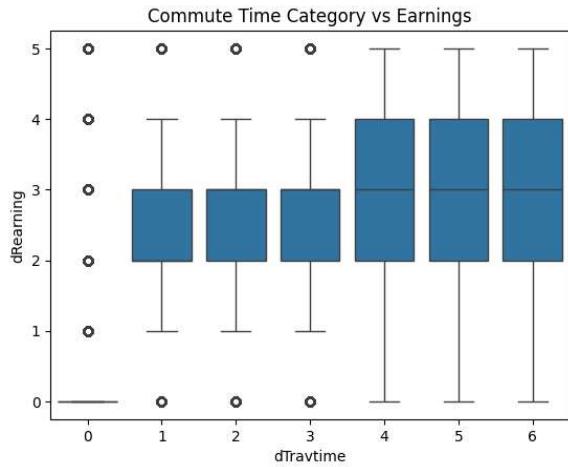


Figure 5. Top 10 Industry Categories



3.3 Bivariate Analysis

- **Commute vs. Age (dAge):** Longer commutes are more prevalent among middle-aged workers but are nevertheless quite uniform in relation to age (Figure 6).
- **Commute vs. Earnings (dEarnings):** Higher earnings are associated with longer commutes, consistent with suburban commuting patterns (Figure 7).

Figure 6. Commute Time vs. Age**Figure 7. Commute Time vs. Earnings**

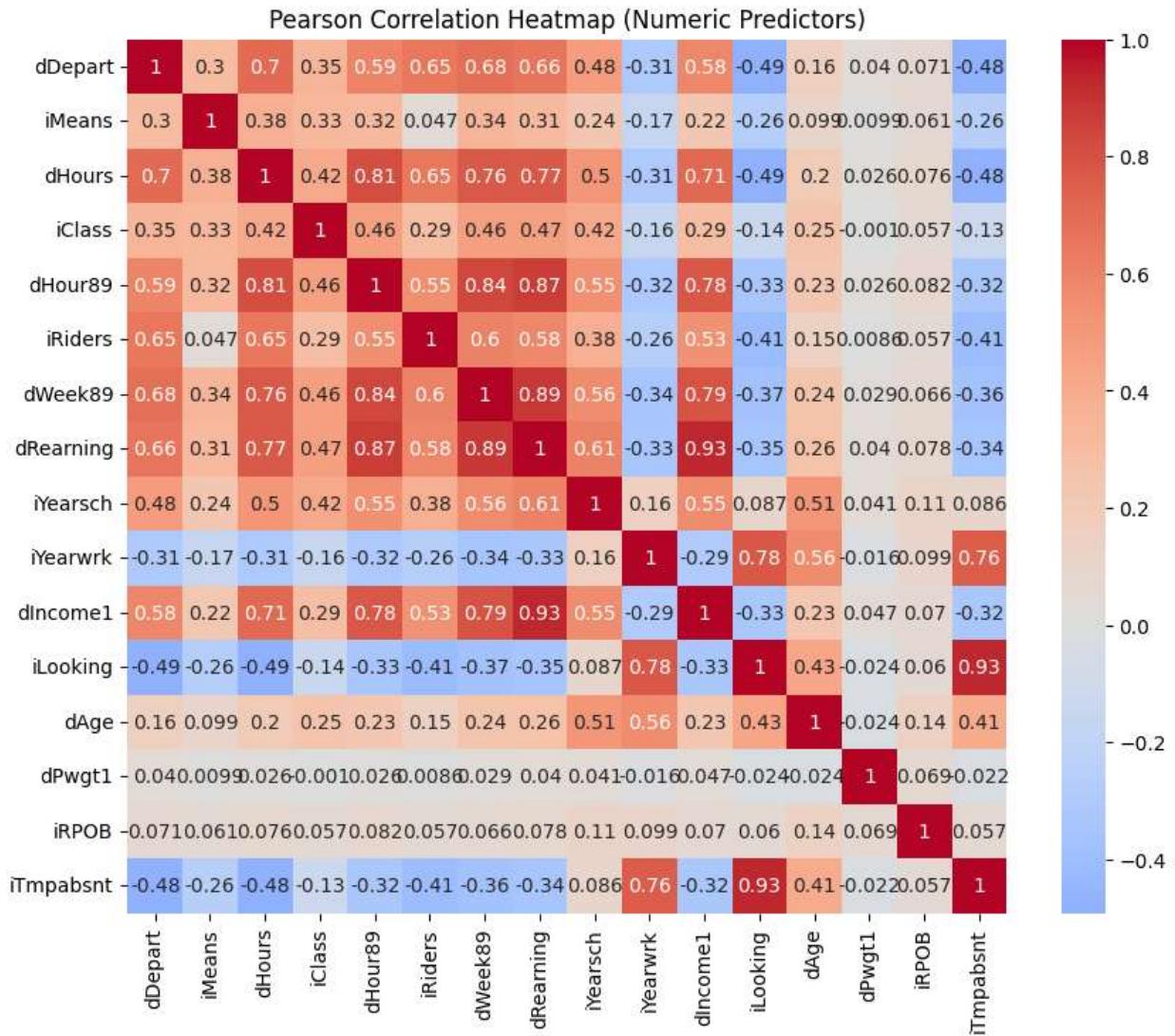
3.4 Statistical Tests

To tease out relationships among covariates and between covariates and the response (dTravtime), statistical tests were conducted. Using ANOVA, all numeric predictors (e.g., dDepart, dHours, dIncome1) exhibited statistically significant differences across commute categories ($p < 0.001$). Using the Chi-square test, categorical predictors (dOccup, dIndustry, iWorklwk) were strongly associated with commute time ($p < 0.001$).

3.5 Correlation Structure

Pearson correlation analysis (Figure 8) reveals strong correlations among work and earnings variables (dHours, dRearing, dIncome1). While this seems obvious, it is satisfying that this relationship exists since it confirms the internal validity of the dataset. It also shows moderate correlation between schooling (iYearsch) and years worked (iYearwrk). We see negative correlation between unemployment indicators (iLooking) and income, as one would expect.

Figure 8. Pearson Correlation Heatmap (Numeric Predictors)



4. Modeling Approach

Given the large size and complex nature of the data, a fair amount of preprocessing is required to manage computational cost and allow the unique solution of multinomial regression. Numeric variables were standardized, and categorical variables were one-hot encoded.

The majority of observations landed in class 0 for dTravTime. Therefore, to improve model performance, classes had to be balanced for model training. Undersampling applied to equalize commute categories. To curb computational cost, each model worked off of a sample of either 50,000 or 100,000 observations as opposed to 80% of the full data set.

This was done to allow models to finish computing in a reasonable time frame, without sacrificing model accuracy.

- **Models evaluated:**
 - Multinomial logistic regression (with backward stepwise selection).
 - KNN (with principal component analysis).
 - Random forest.
 - XGBoost.
 - SVM (RBF kernel with PCA).
 - **Evaluation metrics:** Accuracy, macro-averaged F1 score, and ROC curves.
-

5. Results

5.1 Logistic Regression (Stepwise)

Although backward selection typically starts with a full model (all covariates included), collinearity and multi-factor variables (once categorical variables were one-hot encoded) caused the design matrix to not be full rank, and therefore a full model was not uniquely identifiable. Therefore, the “full model” included 20 variables, identified as those covariates most highly correlated with dTravtime, using the statistical tests completed previously.

Backward stepwise selection identified eight predictors (dDepart, dIncome1, dIndustry, iMeans, iMilitary, dOccup, iPerscare, dPoverty) as optimal. The resulting model achieved roughly 35% accuracy (Figure 9). ROC curves demonstrated moderate separation across classes (Figure 10). Class 0 for dTravTime was by far the easiest to predict. However, there is no clear standout of other easier classes that the multinomial logistic model could successfully predict. This would not hold true for other models.

Figure 9. Backward Stepwise Accuracy vs. Predictors Retained

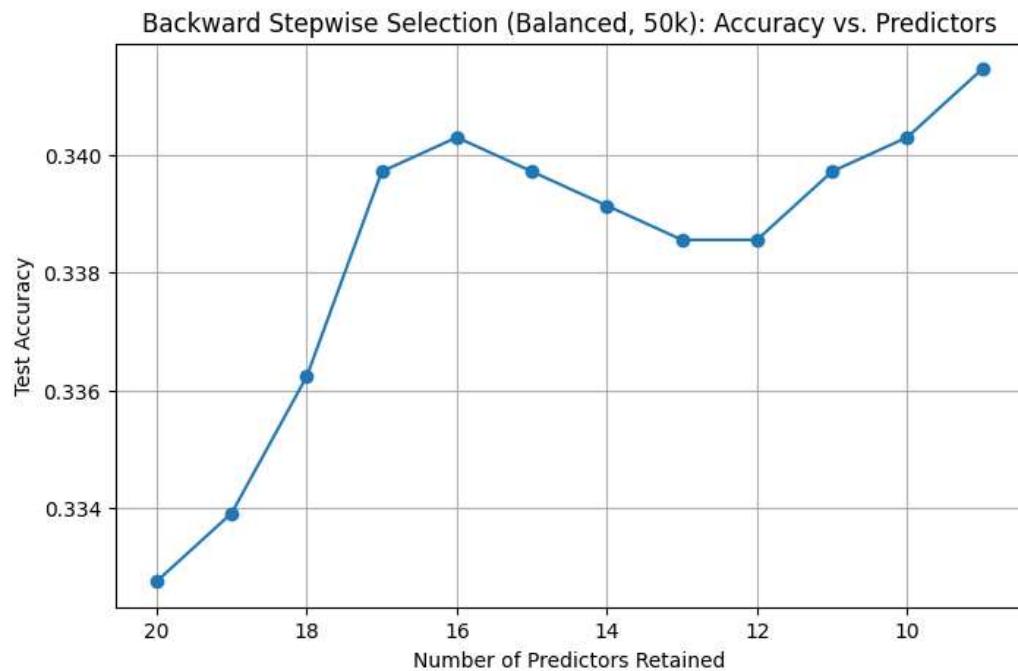
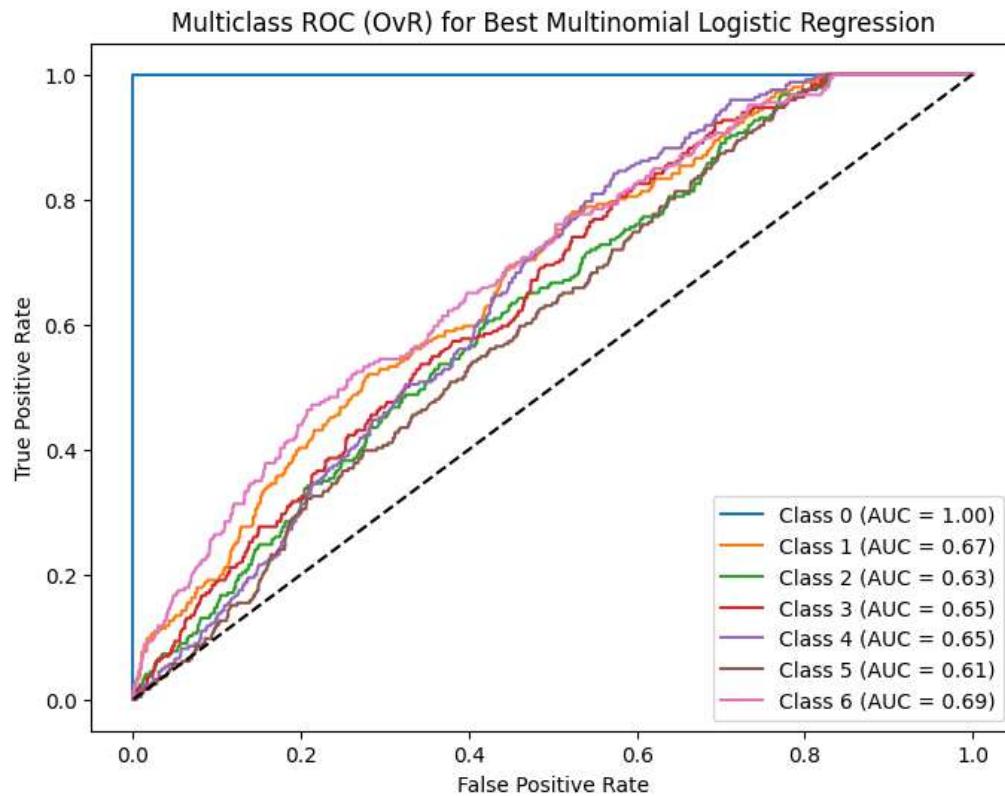


Figure 10. ROC Curves for Best Logistic Regression Model

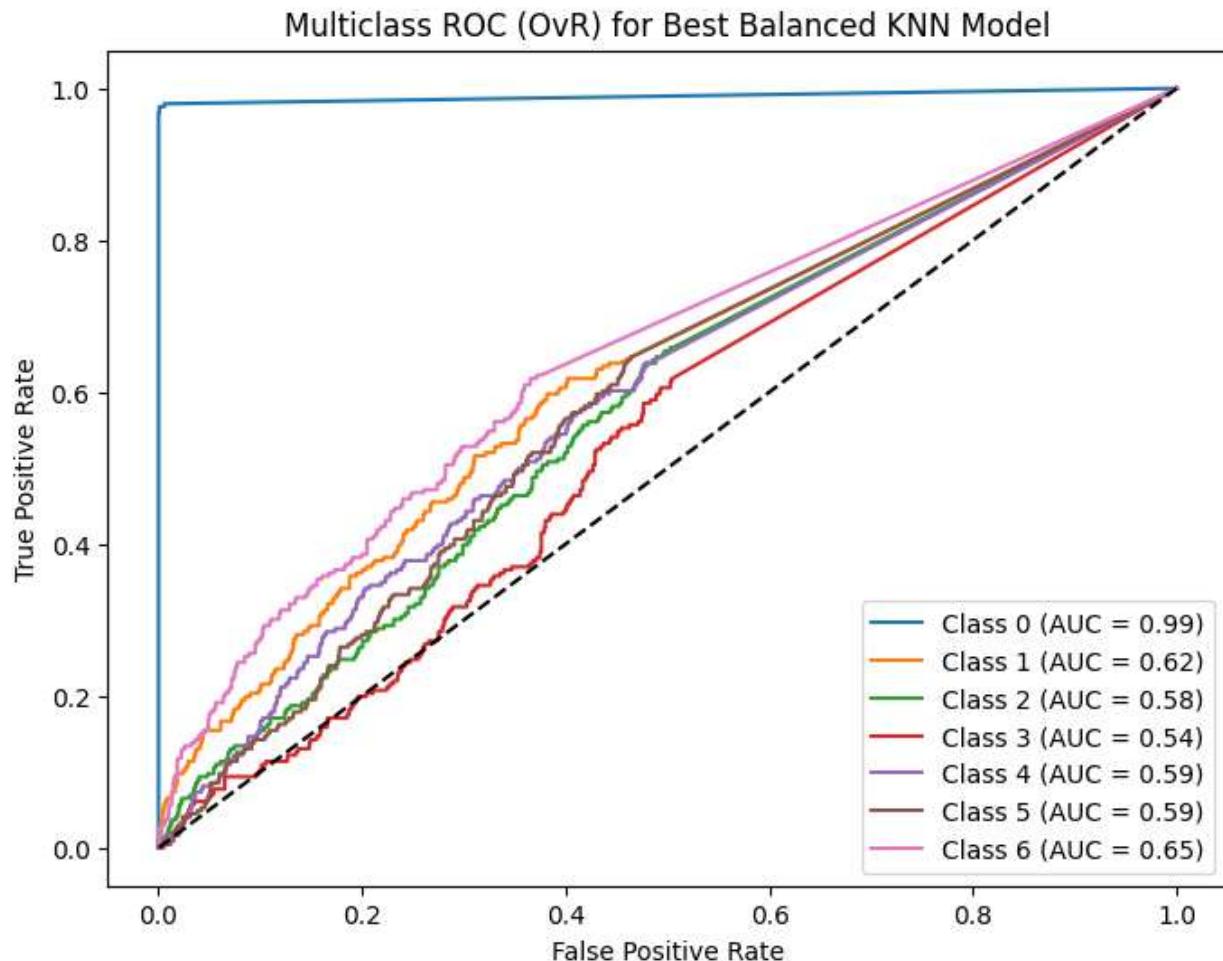


5.2 K-nearest-neighbors

In the k-nearest-neighbors (KNN) model, the same standardization procedure was applied. This procedure was applied across all models in this study. However, to avoid the curse of dimensionality, Principal Component Analysis (PCA) was performed for KNN. PCA retained 95% of variance. These components were used to train the model.

Again, undersampling ensured equal representation across commute time categories at roughly 1,200 observations per category. The models were fit with varying neighborhood sizes ($k = 3, 5, 7, 9$) to tune the hyperparameter. Performance was evaluated using accuracy and macro-averaged F1 scores. After evaluating, $k = 5$ was selected as a representative configuration.

Figure 11. ROC Curves for Best KNN Model



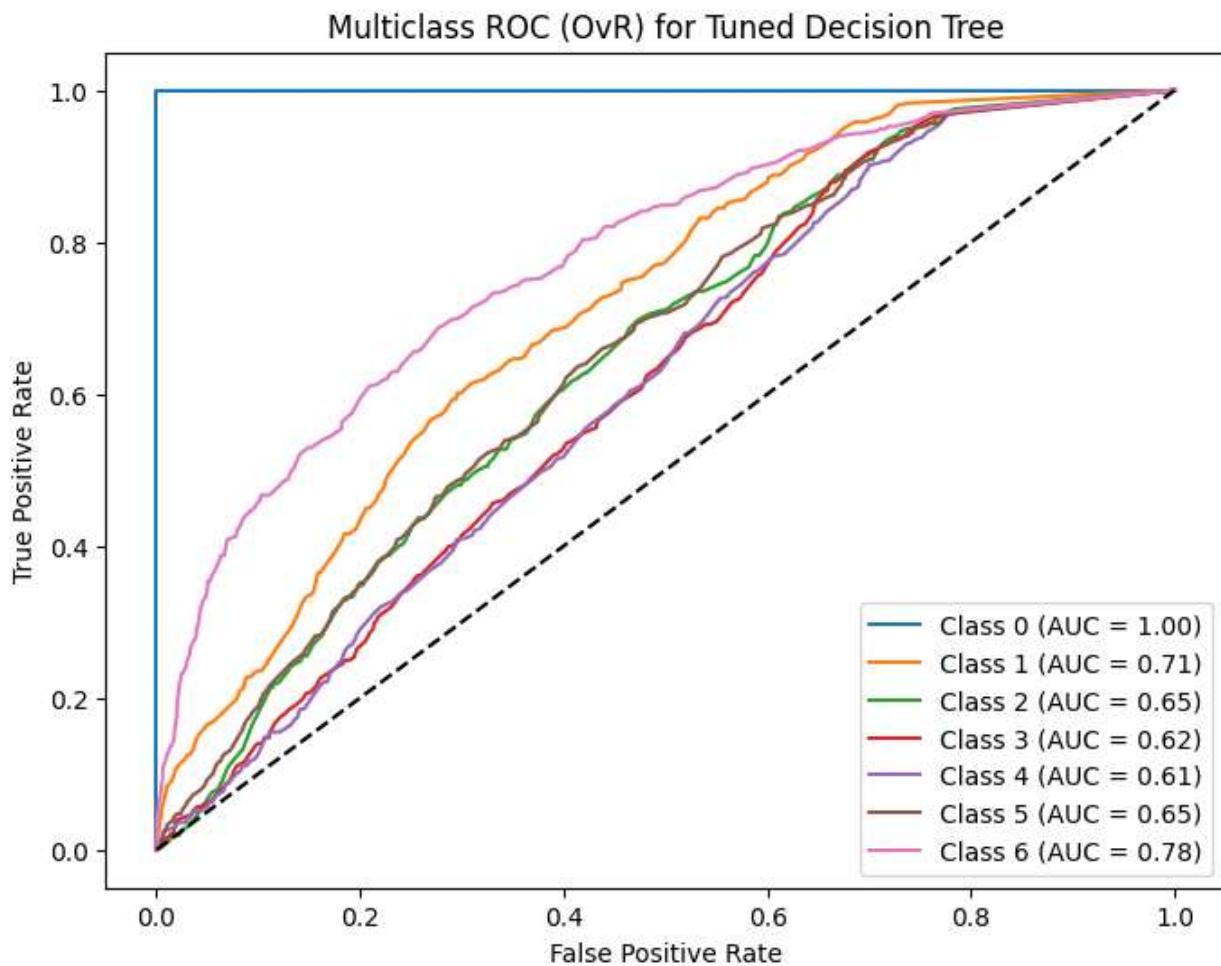
This model yielded approximately 31 percent accuracy and a macro F1 of 0.311, the lowest among the models evaluated. The AUC for each class besides Class 0 sat between 0.55 – 0.65 (Figure 11). This underperformance reflects the difficulty of applying distance-based

methods to high-dimensional, discretized census data, where Euclidean similarity is less meaningful and class boundaries overlap substantially. Even with PCA, the sparsity introduced by undersampling weakened neighborhood consistency, further limiting predictive power. Overall, KNN proved to be ill-suited for this task.

5.3 Single tree

Although ensemble tree methods were conducted later, first, a tuned decision tree was used to model commute time. It performed markedly better than KNN, with an accuracy score of roughly 36% and AUC approaching 0.8 for Class 6, meaning it was more able to identify long commute times than any model so far (Figure 12).

Figure 12. ROC Curves for Single Decision Tree

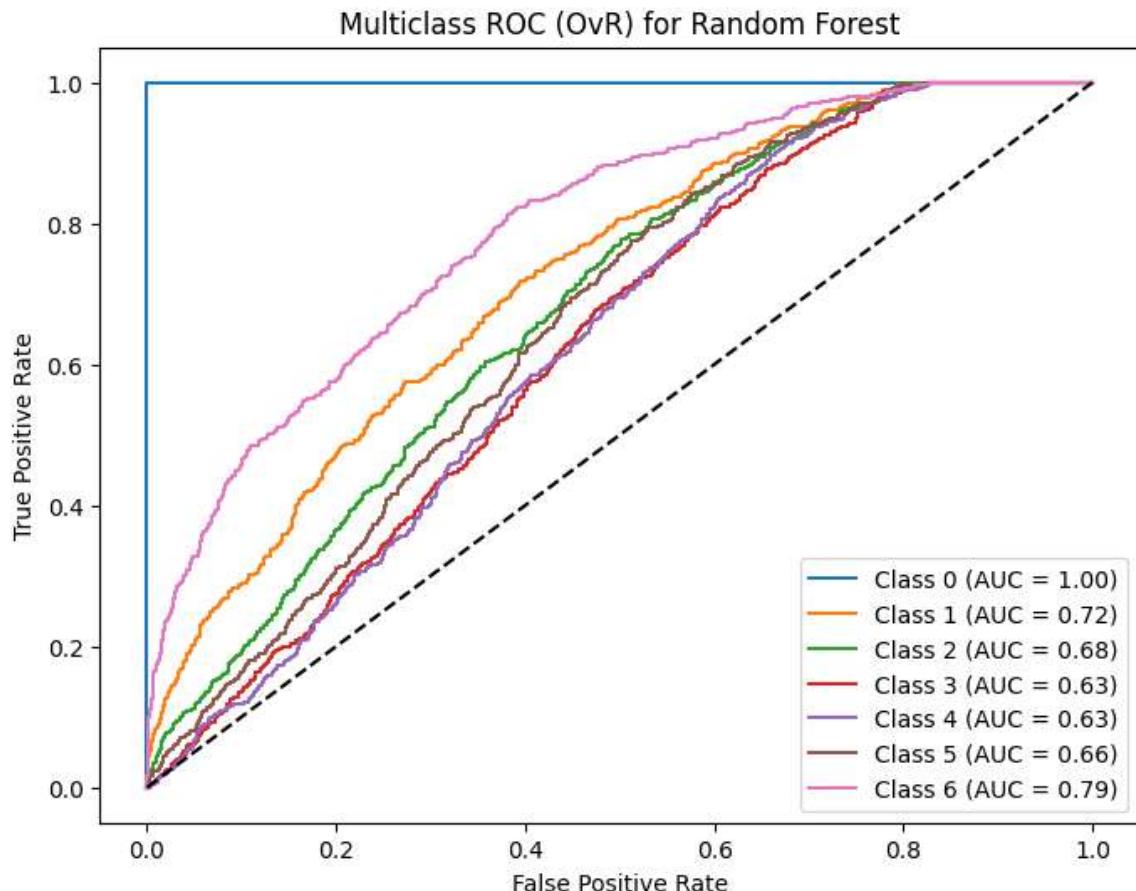


5.4 Random Forest

Hyperparameters were tuned using CV. Evaluation was conducted using accuracy, macro-averaged F1 scores, and ROC curves. The random forest achieved approximately 36.3 percent accuracy and a macro F1 of 0.358, the second-highest among the models tested. It was even better than the single tree, with AUC scores at or above those of the single tree for all classes (Figure 13). This performance reflects the model's ability to capture complex, nonlinear relationships between predictors and commute time categories.

Feature importance analysis highlighted departure time, weekly hours, occupation, industry, and schooling as key drivers, consistent with socioeconomic expectations. While random forest offers strong predictive performance, its interpretability is limited compared to logistic regression, as decision boundaries are aggregated across many trees. Nonetheless, the model demonstrates that ensemble methods are particularly well-suited to high-dimensional census data, where heterogeneous predictors interact in non-linear ways.

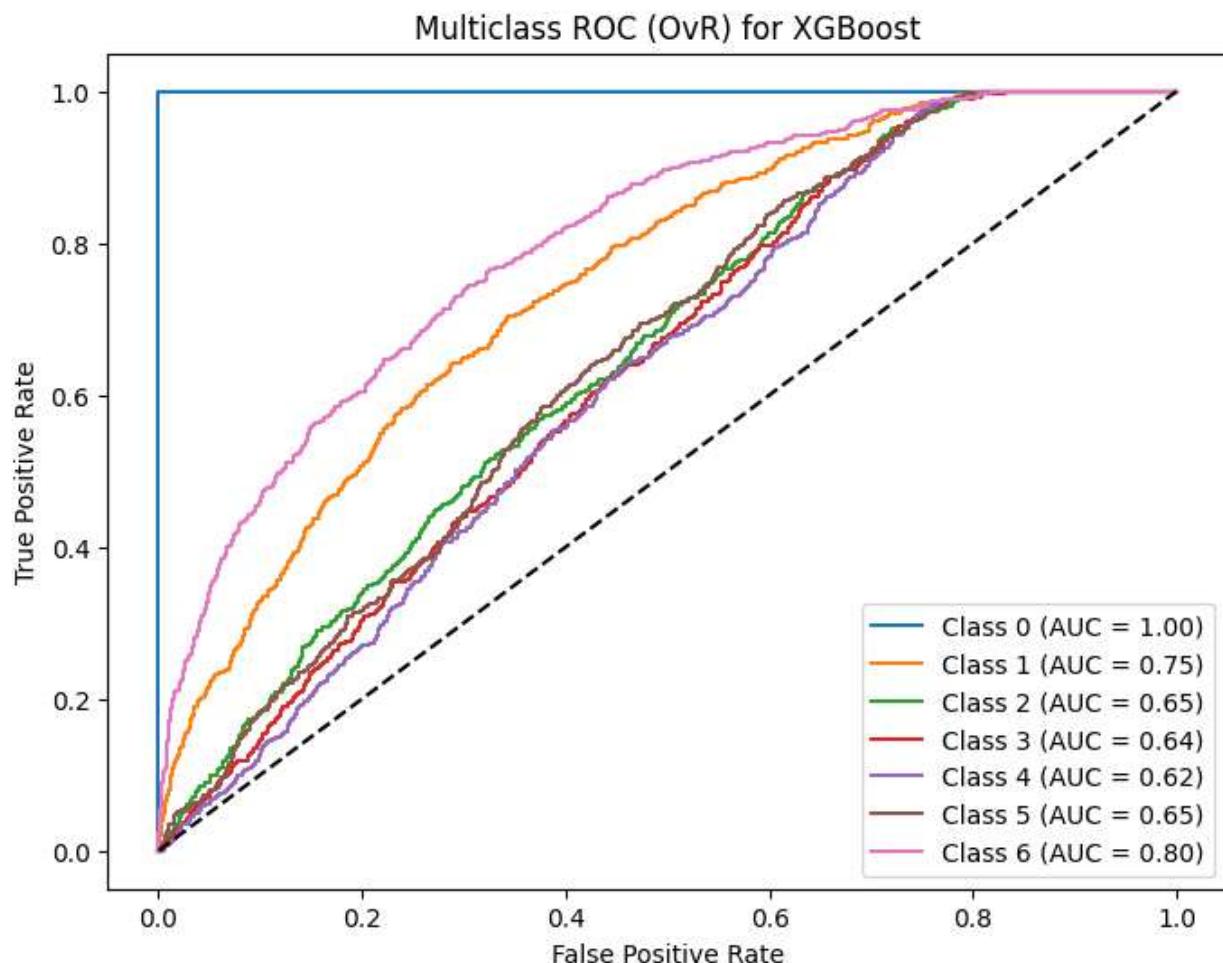
Figure 13. ROC Curves for Random Forest



5.5 Gradient boosting

For the gradient boosting (XGBoost) model, predictors were processed through the same standardized pipeline as the other classifiers, with numeric variables scaled and categorical variables one-hot encoded. Hyperparameters were tuned using CV. The XGBoost model achieved approximately 37 percent accuracy and a macro F1 of 0.367, placing it above other models. Its AUC were higher for both Class 6 and Class 1, meaning it was able to predict long commutes and reasonably short commutes rather well, though it still fell short for intermediate commute lengths (Figure 14).

Figure 14. ROC Curves for Gradient Boosting (XGBoost)

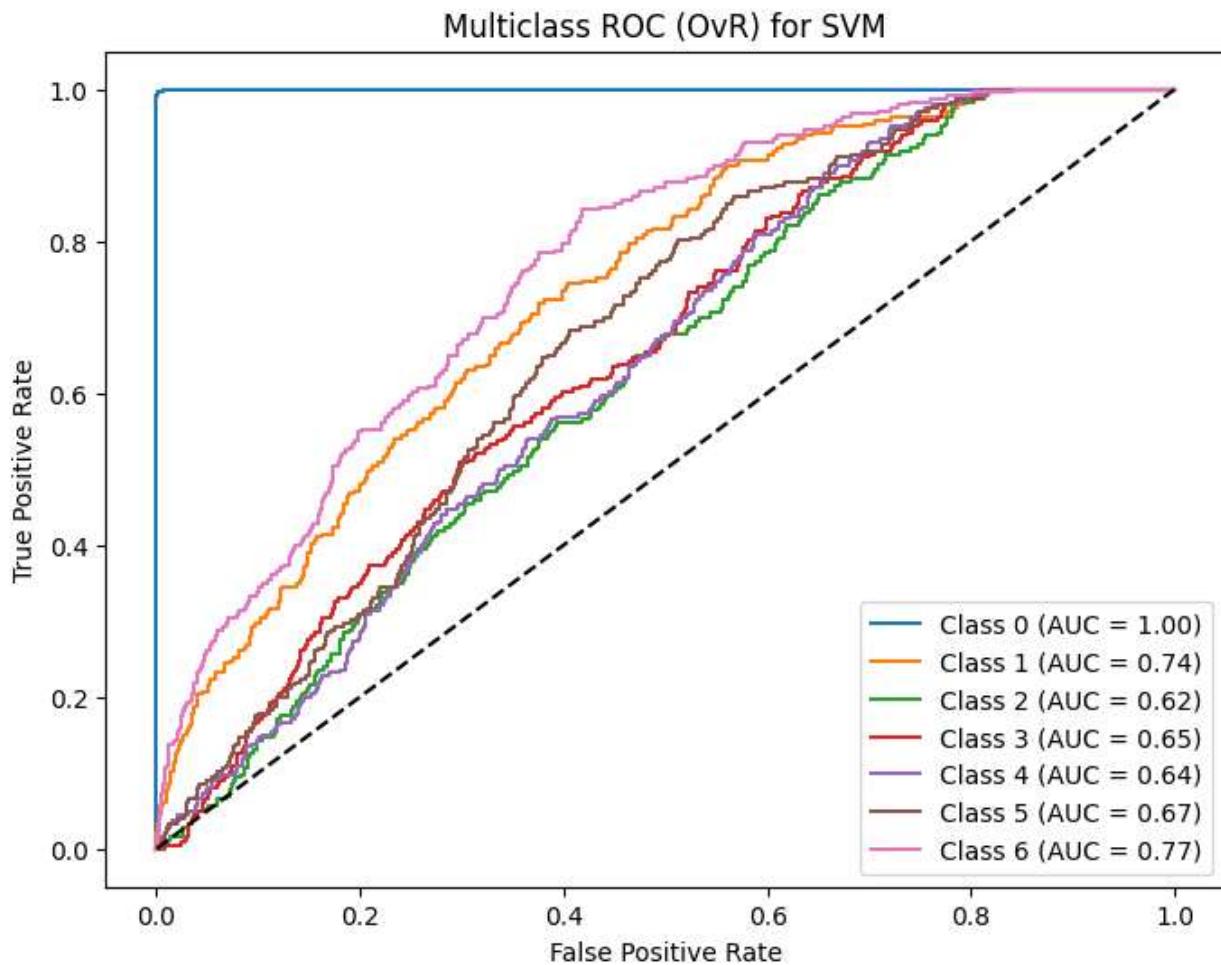


5.6 Support vector machine (SVM)

The support vector machine (SVM) model was tuned using CV and used PCA, as this was essential to reduce the high dimensionality problem. The best SVM model achieved approximately 35.1 percent accuracy and a macro F1 score of 0.344, placing it among the

strongest performers in the study alongside random forest and XGBoost. This level of performance reflects the model's capacity to construct flexible decision boundaries that capture nonlinear relationships in the census data. Unlike logistic regression, which assumes linear separability, the SVM was able to exploit complex interactions between demographic, occupational, and socioeconomic predictors to improve classification. The relatively high accuracy demonstrates that commute time categories, though noisy and overlapping, can be partially distinguished through nonlinear separation in feature space. However, interpretability remains limited. In practice, this makes SVM less transparent than logistic regression or tree-based ensembles, though its predictive strength underscores the value of margin-based classifiers for high-dimensional categorical data.

Figure 15. ROC Curves for Support Vector Machine (SVM)



5.7 Model Comparison

Figure 16. Comparative Performance of Models

Model	Accuracy Macro F1	
Multinomial Logistic Regression (Stepwise)	0.341	0.345
KNN (k=5, PCA)	0.310	0.311
Random Forest	0.363	0.359
XGBoost	0.371	0.367
SVM (RBF, PCA)	0.351	0.344

5.8 Feature Importance

Boosting models highlight the following predictors as most influential (Figure 12):

- Departure time (dDepart)
- Means of transport (iMeans)
- Weekly hours (dHours)
- Occupation (dOccup)
- Industry (dIndustry)
- Work last week (iWorklwk)
- Years of schooling (iYearsch)

These track with our expectations of travel time. People who leave at a certain time are likely to have longer or shorter commutes. The same could be said for people who have the same job or work in the same industry, or who take the same means of transportation.

Figure 17. Top 20 Feature Importances (XGBoost)

Feature	Importance
dIndustry_0	0.095786
dDepart	0.059722
dHours	0.032660
iMeans	0.026468

iRiders	0.018088
dPOB_0	0.013404
iLooking	0.013228
dRearning	0.009775
dIncome1	0.009658
dIndustry_7	0.008729
dPoverty	0.008725
iRlabor	0.008700
dRpincome	0.008302
dIndustry_Other	0.008279
iLang1_2	0.008221
iClass	0.008034
dPOB_3	0.007979
iSchool_1	0.007927
dAncstry1_Other	0.007883
iPerscare	0.007865

6. Discussion

The results underscore the trade-off between interpretability and predictive accuracy. Logistic regression, particularly with stepwise selection, offers transparent coefficient estimates. Tree and tree ensemble methods achieved the highest accuracy (~36%), with XGBoost both leading the way and providing interpretable feature importance.

Commute time is strongly influenced by departure time, occupation, industry, and socioeconomic indicators. However, the discretization of variables into bins limits granularity, and undersampling may reduce generalizability. The modest accuracy levels reflect the inherent complexity and noise in commute behavior.

This also reflects the inherent difficulty in modeling all variables collected in census data. There are many 0's in dTravTime, denoting a 0-minute commute time. This likely comes from people who do not work, given the relatively low prevalence of remote work in 1990.

7. Conclusion

Classical machine learning models achieve approximately 35–36% accuracy in predicting commute time categories. Random forest and SVM are the strongest performers, with logistic regression competitive when optimized.

Future research should explore ensemble stacking, oversampling techniques (e.g., SMOTE), sampling by removing detected non-working individuals, and richer feature engineering (e.g., interaction terms).

Appendix A. UC Irvine SQL Transformations

UC Irvine discretized continuous census variables into categorical bins using SQL functions. Examples include:

- discAge: 0 = age 0, 1 = <13, ..., 7 = ≥ 65 .
- discDepart: 0 = no commute, 1 = <6:00 AM, ..., 5 = $\geq 10:00$ AM.
- disIncome1: 0 = no income, 1 = <15k, 2 = 15–30k, 3 = 30–60k, 4 = ≥ 60 k.
- discTravtime: 0 = no commute, 1 = <10 minutes, ..., 6 = ≥ 60 minutes.

These transformations reduced cardinality, ensured privacy, and facilitated categorical modeling.

Bibliography

Dua, D. & Graff, C. (2017). UCI Machine Learning Repository: US Census Data (1990).
Irvine, CA: University of California, School of Information and Computer Science.
Retrieved from [https://archive.ics.uci.edu/ml/datasets/us+census+data+\(1990\)](https://archive.ics.uci.edu/ml/datasets/us+census+data+(1990))