Income Prediction Report

Darshil Patel (ddp337) and Ayush Sethi (as11500)

Feature Selection, Encoding and Creation

It must be noted that before selecting the final features, many combinations of the features were tried and tested after which the best combination was selected.

To begin, first we looked at the numerical features in the dataset.

Numerical Features:

Apart from our target variable WAGES, we have 5 other numerical features in the dataset. IDNUM, AGE, INTERESTINCOME, TRAVELTIMETOWORK and HOURSWORKPERWEEK. We chose to remove IDNUM from our dataset as it serves no purpose. Then for rest of the features, we replaced '?' values with 0. Choosing 0 was a logical way to impute these values after seeing their explanations.

To see, whether these features will help us in predicting wages, we calculated Pearson Correlation coefficient of all the numerical features with WAGES. From the table below, we can see that INTERESTINCOME is closest to 0(insignificant relation with WAGES). So we chose to ignore INTERESTINCOME from our dataset.

Numerical_Features	Pearson Correlation Coefficient
idnum	-0.006086
age	-0.040988
interestincome	0.002931
traveltimetowork	0.287884
hoursworkperweek	0.489955

Categorical Features:

There are 11 categorical features in the dataset. Let's see one by one, how we dealt with each of them.

1. WORKERCLASS: All the ‘?’ values were replaced by 0 making it a new category. After that one hot encoding was performed.
2. VEHICLEOCCUPANCY: This column had a total of 775 ‘?’ values which accounts for more than 70% of our training dataset, so we chose to remove this feature.
3. MEANSOFTRANSPORT: All the ‘?’ values were replaced by 0 making it a new category. After that one hot encoding was performed.
4. MARITAL: All the ‘?’ values were replaced by 0 making it a new category. After that one hot encoding was performed.
5. SCHOOLENROLLMENT: All the ‘?’ values were replaced by 0 making it a new category. After that one hot encoding was performed.
6. ANCESTRYOFPARENT1: We chose to drop this feature because there were around 230 unique values (hot encoding would lead to too many features). It is unlikely that such a specific information will help us in generalization.
7. DEGREEFIELD: We chose to drop this feature as well. This feature had 174 unique categories and also there were a lot of ‘?’ values in the train set for this feature. Also another feature EDUCATIONALATTAIN was giving the similar information in a broader sense.
8. EDUCATIONALATTAIN: All the ‘?’ values were replaced by 0 making it a new category. After that one hot encoding was performed.
9. WORKARRIVALTIME: This feature had 286 unique categories. They were basically representing time frames of 4 minutes across 24 hours. We created a new feature out of this feature. We divided the classes into following 5 categories.

    No Work - (Work from home/”?”)

    lateNight - 12AM - 6AM

    Morning - 6AM - 12AM

    Afternoon - 12PM - 6PM

    Night - 6PM - 12AM

    This new feature was then one hot encoded and WORKARRIVALTIME was dropped.

10. INDUSTRYWORKEDIN: This feature had 268 unique values, where each class represented a industry and sub-sector of that industry. We decided to create a new feature out of this feature. We binned the classes into just the industries they are representing. This way we ended up with a new categorical feature with 20 categories such as AGR, FIN etc. Then we performed one hot encoding for this new feature and dropped the INDUSTRYWORKEDIN feature.
11. SEX: This feature had only two classes, so we decided not to one hot encode this feature.

This selection and creation meant that we had 75 attributes at the end.

Algorithms

We used linear regression with L1(lasso) and L2(ridge) regularization and Random Forest regressor for this problem.

Regression

The first model we tried for this problem is Linear Regression with L1 regularization.

We went with regularization models, because we wanted to leverage the feature selection capabilities of these algorithms. After our initial feature selection and encodings, we ended up with 75 features for our problem. Since, we had only 1184 examples to train our model, we wanted to work with smaller set of features.

Below are the values of Root Mean Squared Error we achieved with both L1 and L2 regularizations on cross validated alphas. We performed 10 fold cross validation to select the best value of alpha for both the variations of linear regression.

Combination	**RMSE**
L1, Alpha = 100	69810
L2, Alpha = 28	69108

Since we were not able to achieve great results with linear regressors, we decided to try Random forest regressor.

Random Forest Regressor

Since we were not able to achieve impressive results in Linear regression, we focused on Random forest regressor.

To come up with best configuration for our model, we tried below configurations of hyperparameters to come up with best possible model.

HyperParameteres	Values
n_estimators	[200, 400, 600, 800, 1000, 1200, 1400, 1600, 1800, 2000]
'max_features'	['auto', 'sqrt']

'max_depth'	[10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 110]
'min_samples_split'	[2, 5, 10]
'min_samples_leaf'	[1, 2, 4]
‘bootstrap’	[True, False]

We used Randomized Search CV with above parameter ranges. We used a 3 fold CV with 100 iterations, so in total model was fitted 300 times with a different combination each time.

Best configurations we ended up with :

'bootstrap': False,
'max_depth': None,
'max_features': 'sqrt',
'min_samples_leaf': 2,
'min_samples_split': 5,
'n_estimators': 1400

Our experiment found that RMSE decreases as we increase the n_estimators. However, to check for overfitting, we used 10-fold cross validation. The cross validation results shows that for some of the folds, the test error was extremely high (exponential) indicating overfitting. The cross validation results for Random Forest regressor with best parameters are in the table below.

Train RMSE									
Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Fold 6	Fold 7	Fold 8	Fold 9	Fold 10
22870.17	41884.60	31897.72	86472.84	90129.12	41665.66	29687.43	18288.12	66869.56	68516.17

Conclusion

As can be seen from the report, a lot of time in this project went in data preparation. We also tried regression models by taking log of WAGES, because it’s not normally distributed. But no significant improvement was seen. So, for our final model, we chose to go with Random forest regressor. Even though we know that this model still has problem of overfitting, but it is performing better than the linear regression models.