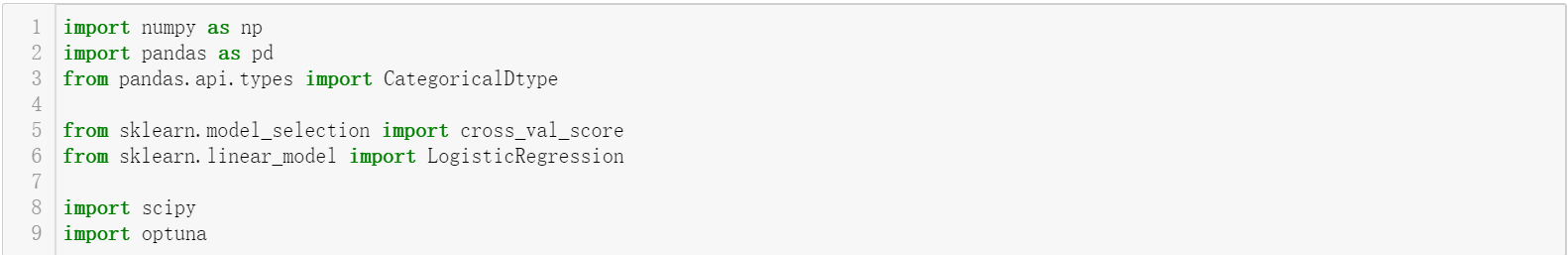
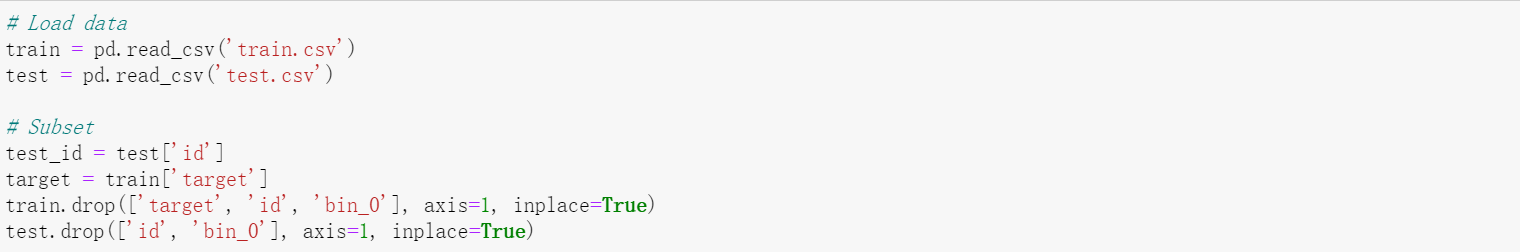
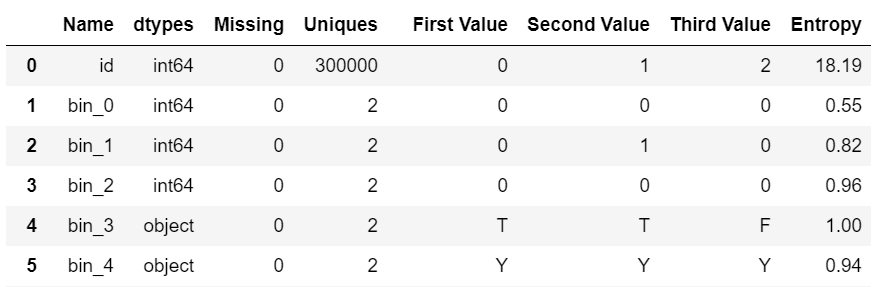
1. Import libraries.

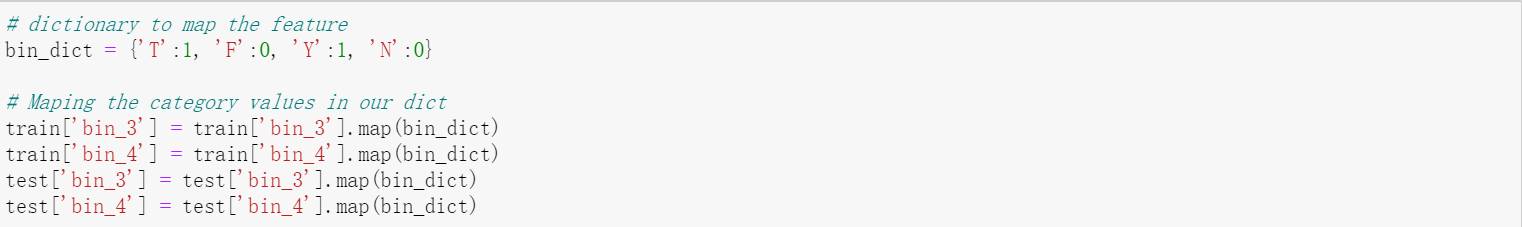


1. Load the dataset

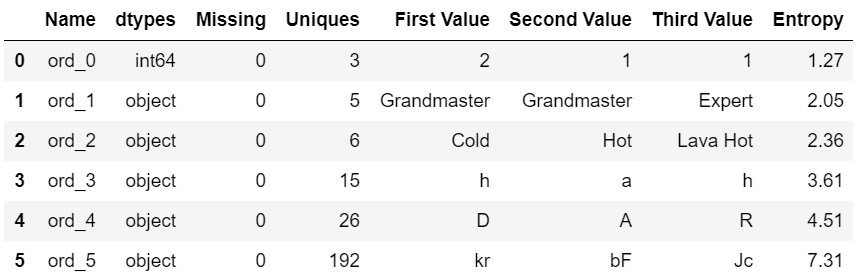


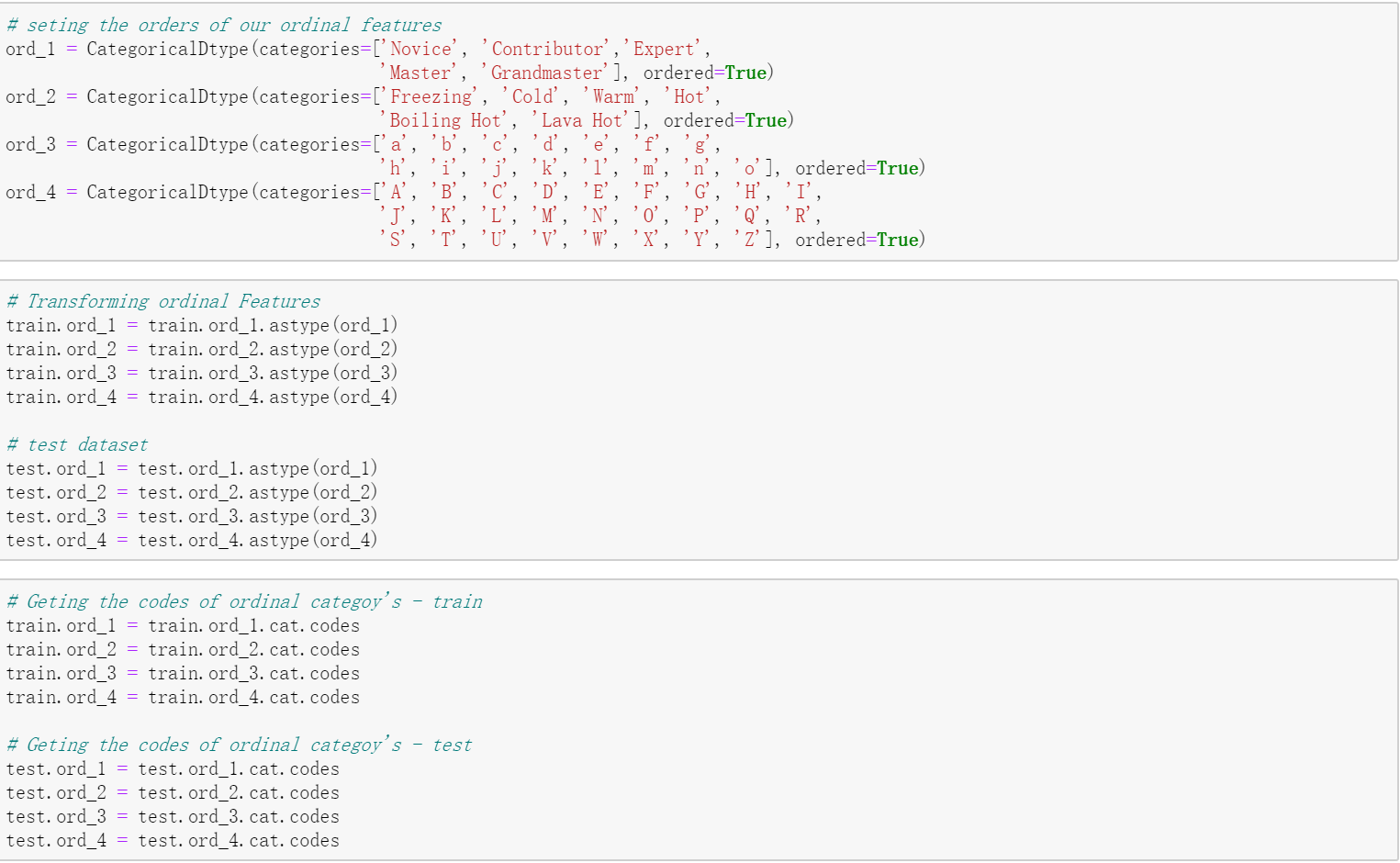
1. As we find in our EDA, the binary feature bin\_3 & bin\_4 are not numeric data. Therefore, we will map them to int format based on assumption that, T/Y is 1 and F/N is 0.



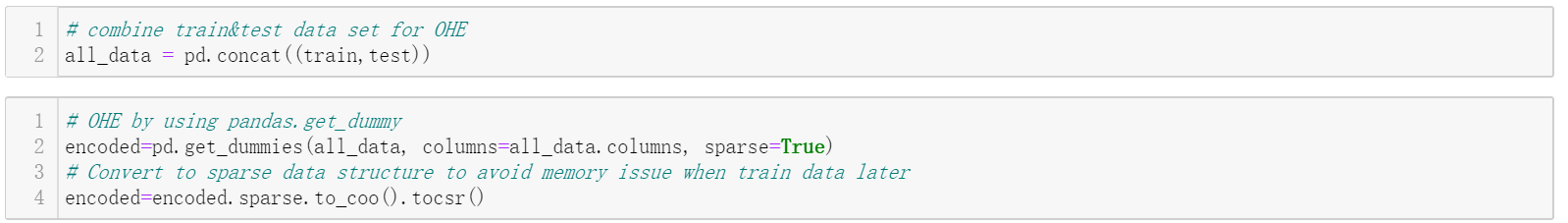


1. For ordinal features, ord\_1 to ord\_4, based on our finding from EDA. They have fair number of unique values. We could use label encoding by mapping their unique value with an integer. The Ord\_5 has unique value up to 192, that is far too many for doing a mapping. We will pass that, and one hot encoding it. Ord\_0 is already numeric feature; we will leave it like that.

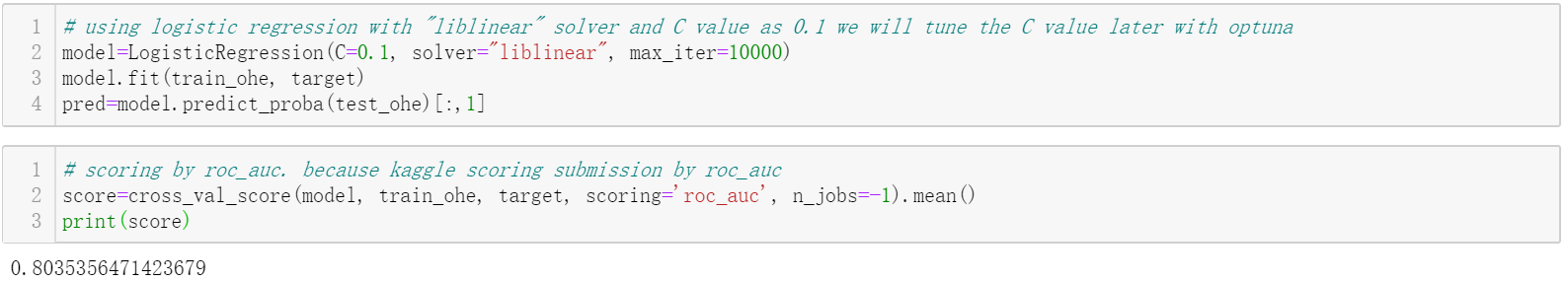




1. The rest of the feature, we use one hot encoding to convert. For overcome the memory issue which we met before. We convert the data to sparse data type.



1. Train data by using Logistic Regression.

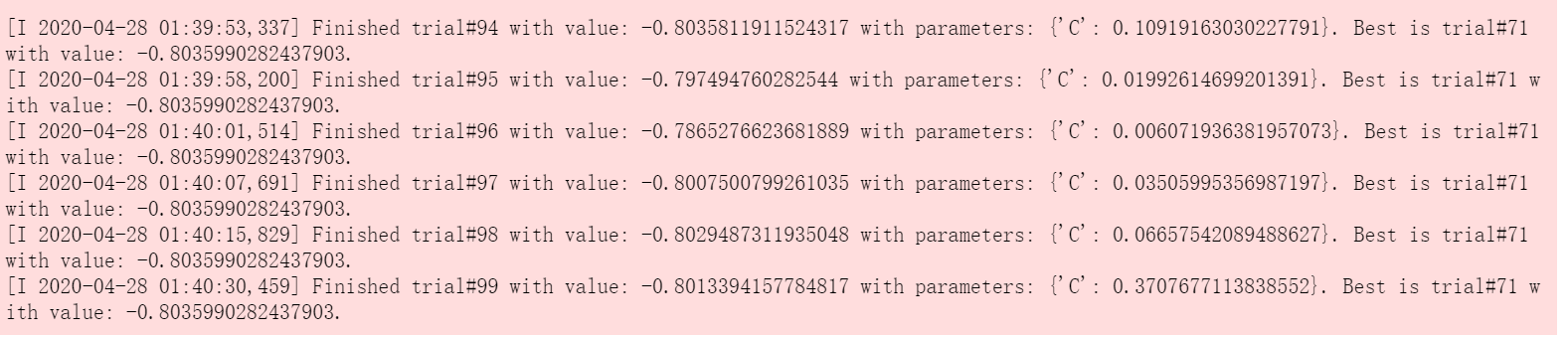


1. Tune C value by using optuna. (An open source hyperparameter optimization framework to automate hyperparameter search)

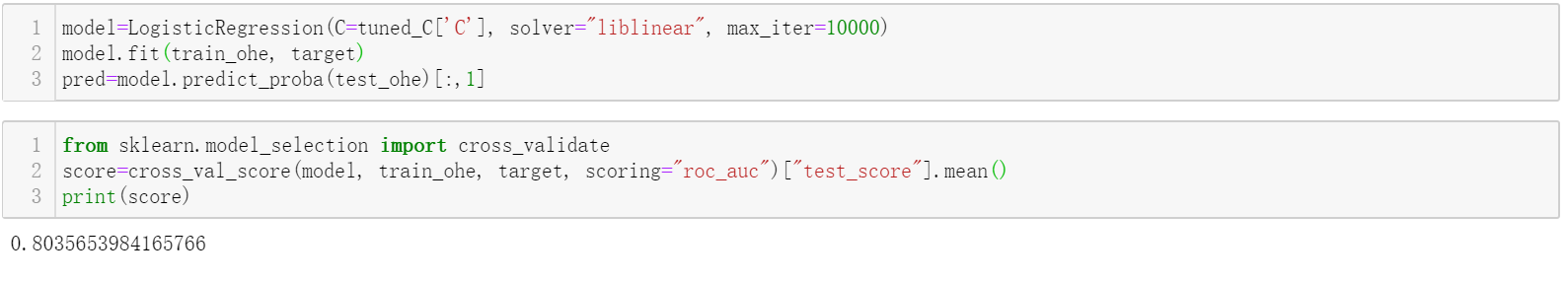
The optuna trying different C value from 10e-10 to 10, run the same evaluation function to find the best C value.

After 100 trial, we get our best C value from trail #71.





1. Train data again with best C. Score is slightly better than before.



1. Submit to Kaggle.

