HackerEarth Machine Learning challenge: Adopt a buddy

Data - :

- Train consisted of 18834 and test had 8072 rows. Raw dataset contained 8 features excluding pet_id column.

Problem - :

- This dataset is a type of multioutput multiclass classification i.e. there are two target variables and both of them has more than two levels.

Strategy - :

- So there were three ways in which I could have solve this problem…
    1. Modelling each target variable separately and then predicting both of them separately and combining them in same dataset to check the score.
    2. Modelling each target variable one by one i.e. modelling one target variable alone and using the first one to model the second variable. And for prediction using the output of first variable to predict the second one.
    3. Modelling both the target variables together.
- After reading all the online materials and getting to know the drawbacks of first and second method, I decided to go with the third method.

Models Used - :

Note that not all classifiers support multioutput-multiclass tasks. Below given four base models which support this task were used to check the scores.

- Base Models - :
    1. DecisionTree
    2. RandomForestClassifier
    3. ExtraTreeClassifier
    4. ExtraTreesClassifier

Also multioutput classification support can be added to any classifier with MultiOutput-Classifier. This strategy consists of fitting one classifier per target. This is a simple strategy for extending classifiers that do not natively support multiple-target classification.

- MultiOutputClassifier with - :
    1. DecisionTree
    2. RandomForestClassifier
    3. ExtraTreeClassifier
    4. ExtraTreesClassifier
    5. GradientBoostingClassifier
    6. AdaBoostClassifier with base estimator as RandomForestClassifier

Feature Engineering - :

- Missing column ‘condition’ was imputed with the most frequently occurring value in it i.e. 1.0.
- Generated new feature called days_gap with precision to the seconds using difference of listing_date and issue_date.
- Performed one hot encoding on all the categorical columns i.e. color_type, X1 and X2.
- There were some unique values of color_type and X1 which were not in test data. Therefore, that respective dummy columns had to be dropped from the train data.
- After doing all the feature engineering final dataset had 83 features.

Model Building - :

- So instead of doing train-test split of train data, I preferred training the model on whole train dataset as in most of the cases this technique gives a better result.
- Scores on respective models…

1.	DecisionTree	:	86.43342
2.	RandomForestClassifier	:	89.16559
3.	ExtraTreeClassifier	:	86.00689
4.	ExtraTreesClassifier	:	88.66406
5.	MOC (DT)	:	86.69581
6.	MOC (RF)	:	89.52249
7.	MOC (ETreeC)	:	85.95784
8.	MOC (ETreesC)	:	89.03398
9.	MOC (GBCL)	:	89.20215
10.	MOC (ABRF)	:	89.35583

- From the above scores, we can see that MultiOutputClassifier with RandomForest-Classifier as estimator gives best result. Therefore, this model was finalized and the csv file of this model was submitted as final submission.

Things that didn’t work out - :

- Tried imputing condition column using groupby on color_type but it was resulting in less score.
- At first, in new feature days_gap, I was just taking difference of years.
- Also tried label encoding but dummies were performing better.

Source for MultiOutputClassifier - :

- https://scikit-learn.org/stable/modules/multiclass.html
- https://scikit-learn.org/stable/modules/generated/sklearn.multioutput.MultiOutputClassifier.html