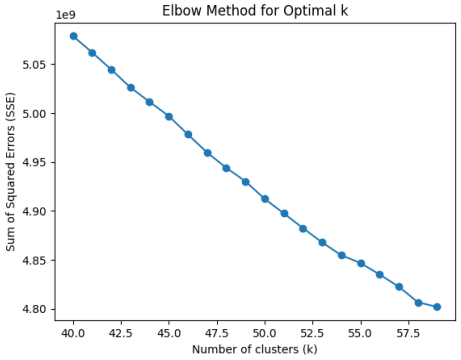
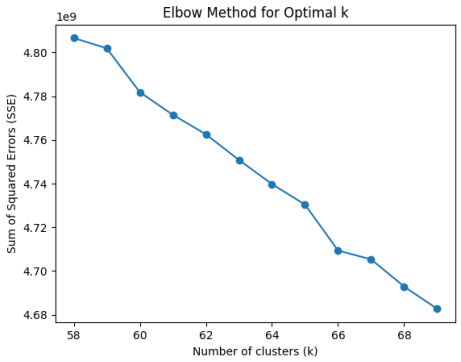
HKUSTGZ-AIAA-2205-HW1-Fall-2024

Chang XU 50012819

Attempts

1. Try running the samples. Try to run the samples and combine three results. For each row, choose the majority result of the three as the result to submit. BAD score: 0.35961
2. Try to find the best methods for this problem. Directly submit mlp, better. score: 0.41228
3. Try add activation='relu', solver='adam', alpha=0.0001, in the MLPClassifier. slightly better score: 0.41315. Try larger iterate times up to 800. BAD score 0.40789
4. Use Elbow Method, find k = 58 or k = 66 clusters. Retrain MLP. better score 0.44736





5. Use Grid Search to search for best hyperparameters for mlp model, and use k-folds cross validation to validate. I get:
   Best Parameters: {'activation': 'relu', 'alpha': 0.001, 'hidden_layer_sizes': (500,), 'learning_rate_init': 0.001, 'solver': 'adam'}
   Best Score: 0.4667844522968198
   In this step, I make the mlp to reach the score of 0.45000 in the private board for k = 66, and reach the score of 0.45614 for k = 100.
6. Try boosting: each step focus on the Residual and use mlp to iterate. This is very bad, only get 0.35263
7. Try XGBoost: Default Parameters get 0.47368
8. Change n_estimators to 300, Change k clusters to 430, get the best score 0.50701. During this period, I use Stratified Cross Validation which is a better choice for validation of classification missions.

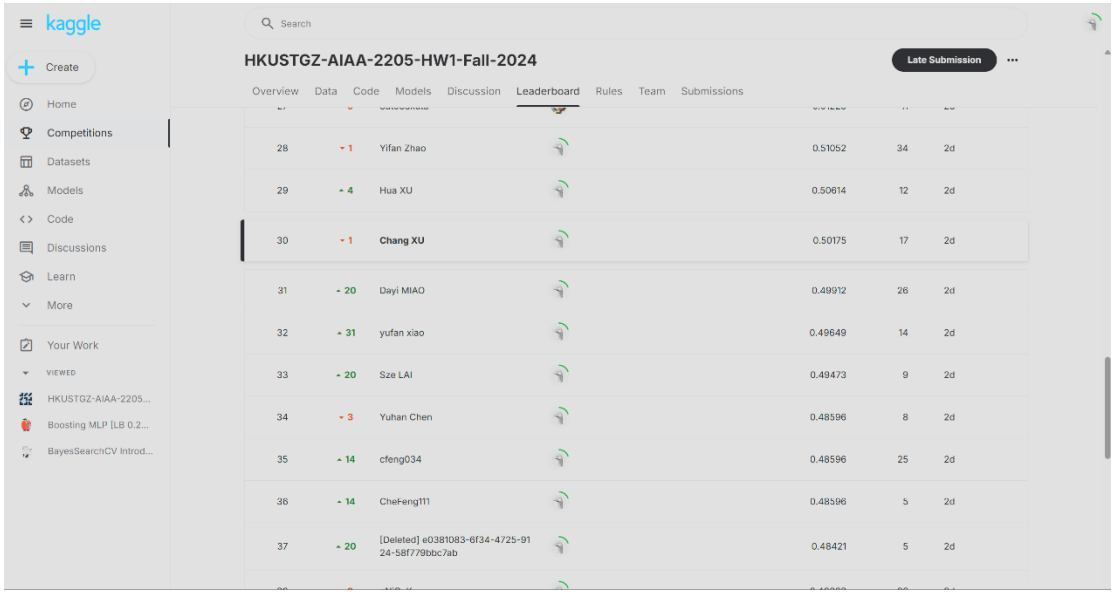
Best Attempt

Use XGBoost, use k clusters and bof430, change the hyperparameters n_estimators = 300, train and run the model.

Submission and Private Leaderboard

Submission Name: Chang XU

Screenshot:



README

HKUSTGZ-AIAA-2205-HW1-Fall-2024

Chang XU 50012819

Package Required

```
scikit-learn pandas tqdm numpy xgboost
```

Select Frames

Choose 50 percent of the total mfccs.

```
python select_frames.py labels/trainval.csv 0.5 selected.mfcc.csv --mfcc_path mfcc/
```

Train k-means model

As shown in the one-page-write-up, the choice of k = 430 is the most suitable for this task.

```
python train_kmeans.py selected.mfcc.csv 430 kmeans.430.model
```

Feature Extraction

```
python get_bof.py kmeans.430.model 430 labels/videos.name.lst --mfcc_path mfcc/ --output_path bof430/
```

Train XGBoost Model

```
python train_XGBoost.py bof430/ 430 labels/trainval.csv models/mfcc-430.XGBoost.model
```

Use XGBoost Model to predict

```
python test_XGBoost.py models/mfcc-430.XGBoost.model bof430 430 labels/test_for_student.label mfcc-430.XGBoost.csv
```

Validation

In this project, I use k-fold cross validation and stratified cross validation, for mlp model test and XGBoost method validation respectively. For mlp validation, the following command are used:

```
python Cross_Validation_MLP.py bof100/ 100 labels/trainval.csv models/mfcc-100.mlp.model
```

Or one can change the numbers in "bof100", "100", and "models/mfcc-100.mlp.model" for different k. For XGBoost validation, use the following cammand:

```
python Stratified_Cross_Validation.py bof430/ 430 labels/trainval.csv models/mfcc-430.XGBoost.mlp.model
```

Also change the corresponding numbers is acceptable.