Classifying Urban sounds using Machine Learning and Deep Learning

3 Model Training and Evaluation

Before training the model, I will convert the label in dataTrain_Validation and dataTest from categorical array to string array. It is because categorical array does not accepted by code generation using MATLAB coder in deployment

```
dataTrain_Validation{:,15}=string(dataTrain_Validation{:,14});
dataTrain_Validation(:,14)=[];
dataTrain_Validation.Properties.VariableNames{14} = 'Label';

dataTest{:,15}=string(dataTest{:,14});
dataTest(:,14)=[];
dataTest.Properties.VariableNames{14} = 'Label';
```

Train the model Using Classification Learner APP

Interactively train, validate, and tune classification models

Choose among various algorithms to train and validate classification models for binary or multiclass problems. After training multiple models, compare their validation errors side-by-side, and then choose the best model. To help you decide which algorithm to use, see Train Classification Models in Classification Learner App.

```
classificationLearner
```

Taking into Consideration of Choosing the Right Model for you.

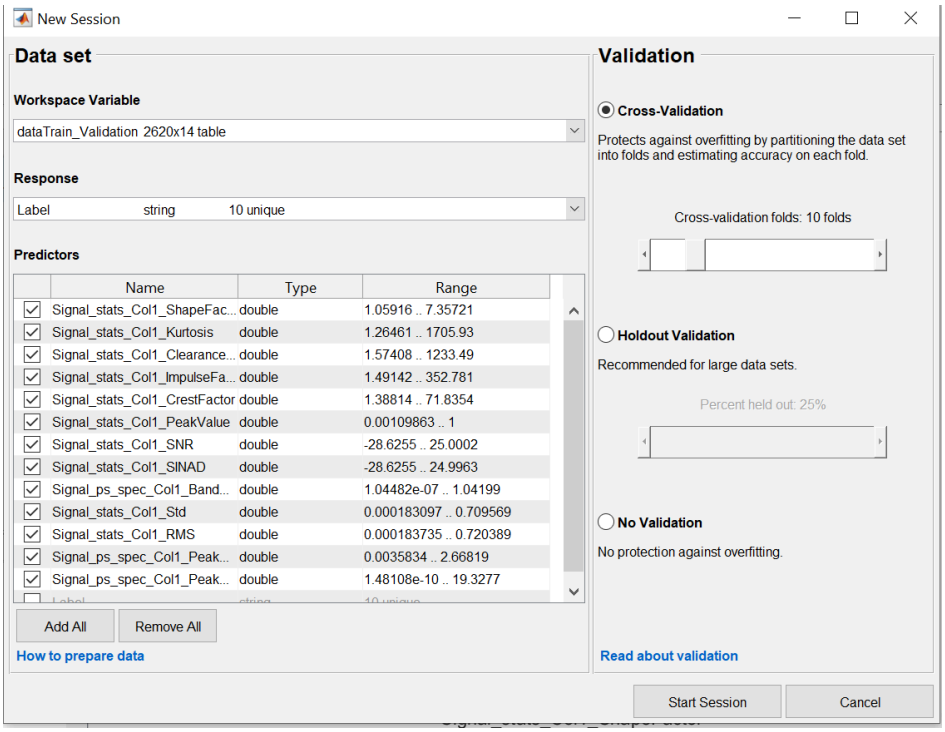
https://www.mathworks.com/help/stats/choose-a-classifier.html

Characteristics of Classifier Types

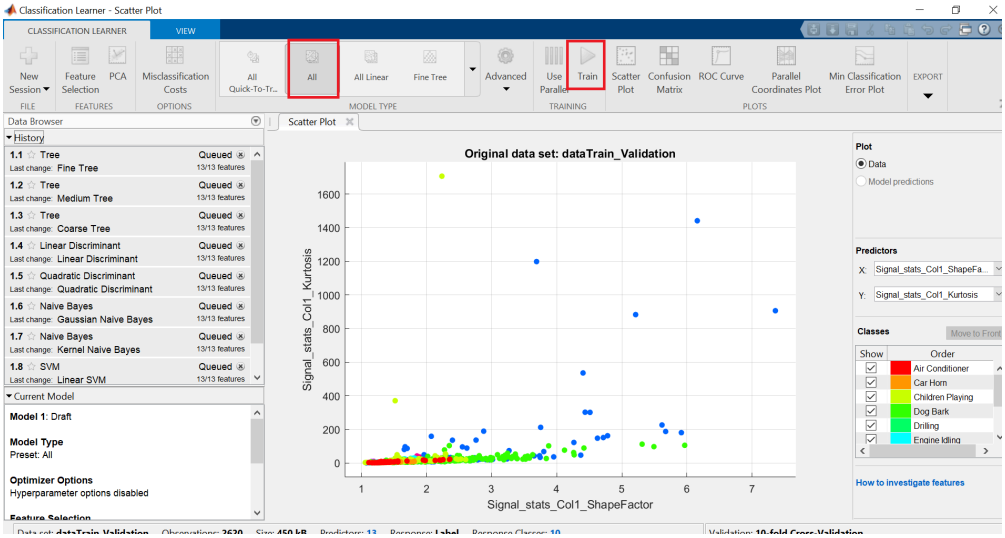
Classifier	Prediction Speed	Memory Usag
Decision Trees 	Fast	Small
Discriminant Analysis 	Fast	Small for linea quadratic
Logistic Regression 	Fast	Medium
Support Vector Machines  **Note** Supports C code generation for prediction.	Medium for linear Slow for others	Medium for lin All others: mec multiclass, larg
Nearest Neighbor Classifiers 	Slow for cubic Medium for others	Medium
Ensemble Classifiers 	Fast to medium depending on choice of algorithm	Low to high de choice of algor
Naive Bayes Classifiers 	Medium for simple distributions Slow for kernel distributions or high-dimensional data	Small for simpl Medium for ke or high-dimens

Training Model in Classification Learner App

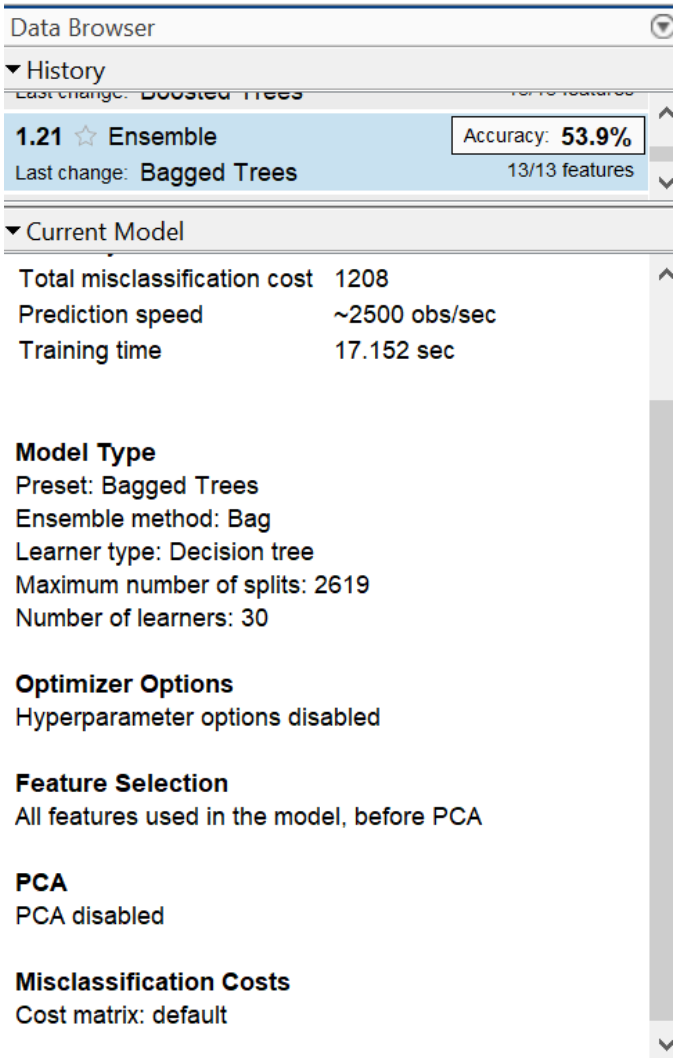
1) Import Data with 10 folds



2) Train All Model

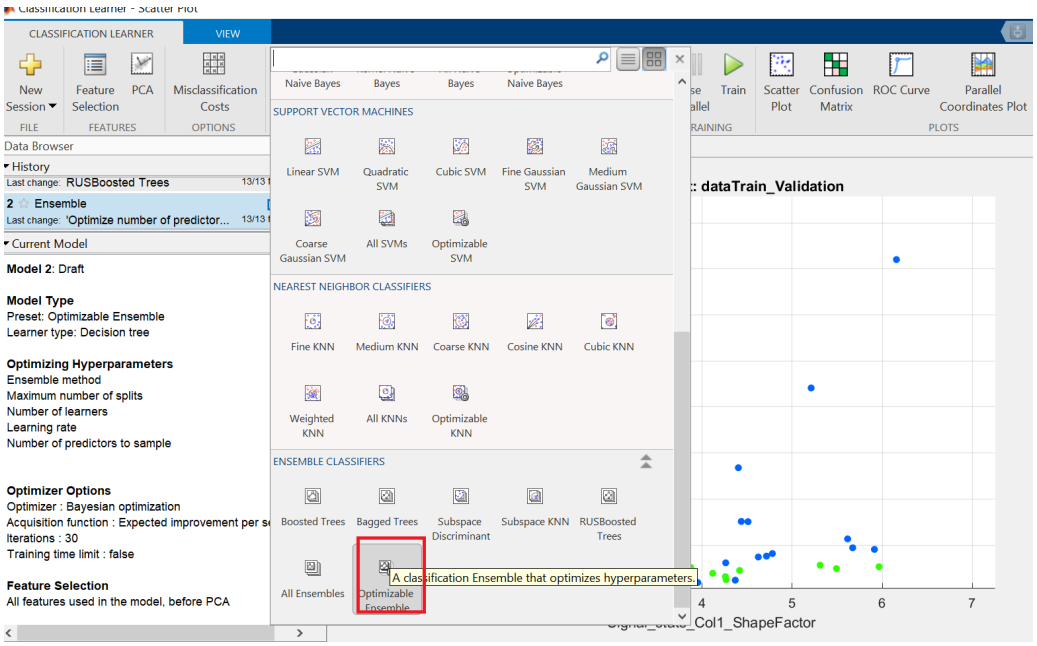


3) First run result, I found bag tree is the best model.

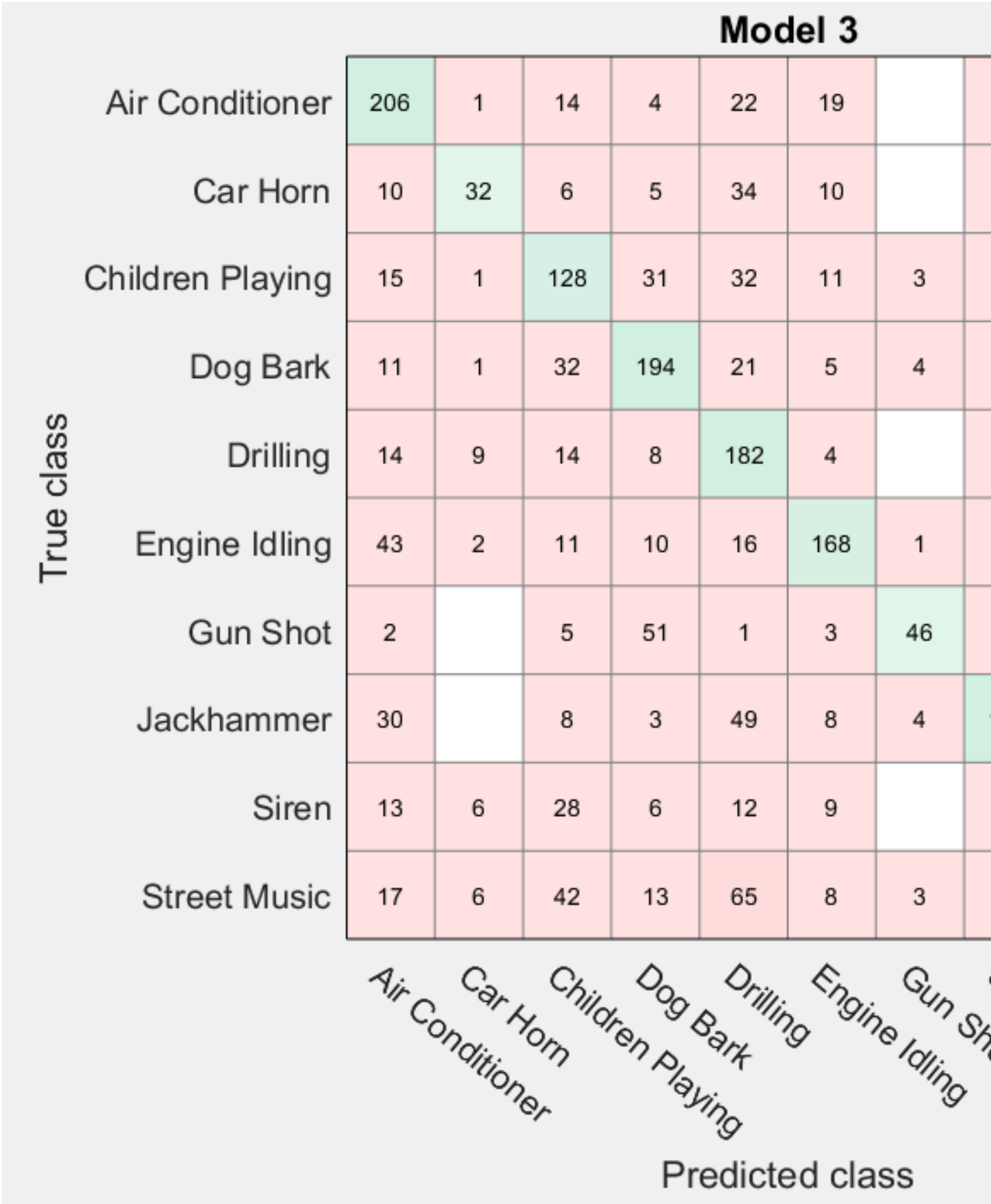


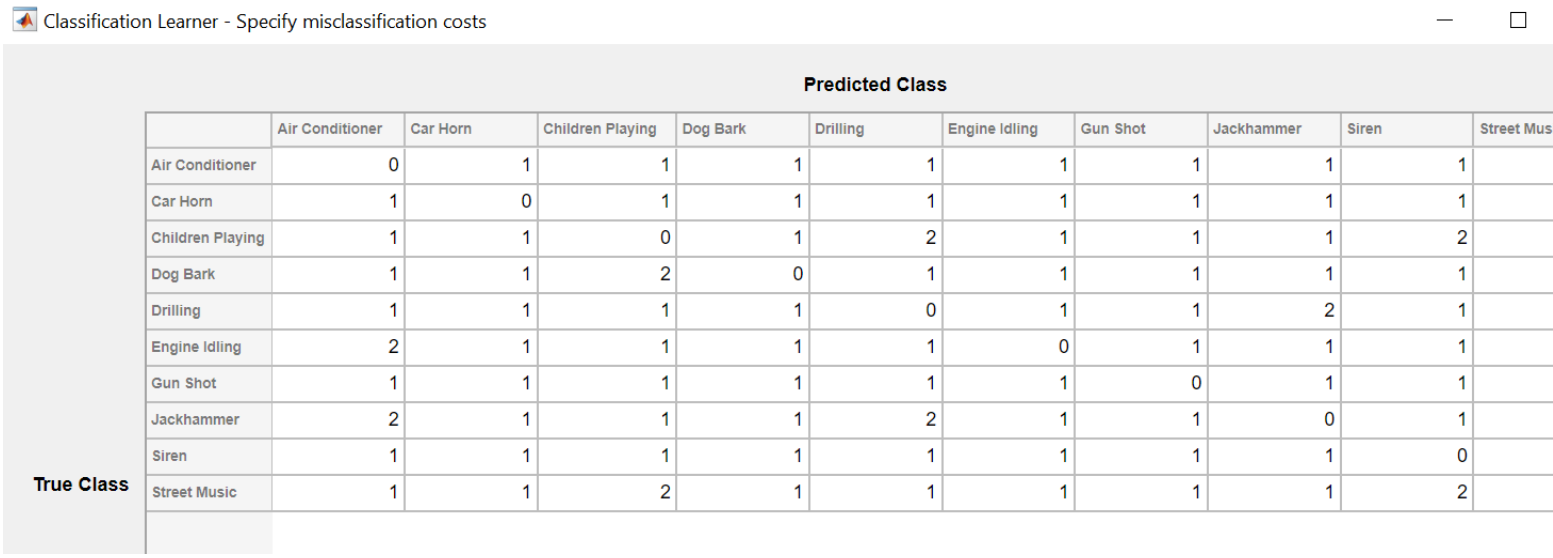
4) Now, we try to use bayesian optimization to optimize the model. (play around with the hyperparameters, see whether can you get higher accuracy model).

Please take note, try to matain the simplest nmodel with high accuracy, else big model might cause us problem in code generation.



5) After Bayesian optimization, I still could not get model with better accuracy. Therefore, i try to impose higher misclassification costs on those wrong prediction more than 30.


Model 3
Air Conditioner
Car Horn
Children Playing
Dog Bark
Drilling
Engine Idling
Gun Shot
Jackhammer
Siren
Street Music
206 1 14 4 22 19
10 32 6 5 34 10
15 1 128 31 32 11 3
11 1 32 194 21 5 4
14 9 14 8 182 4
43 2 11 10 16 168 1
2 5 51 1 3 46
30 8 3 49 8 4
13 6 28 6 12 9
17 6 42 13 65 8 3
True class
Air Conditioner
Car Horn
Children Playing
Dog Bark
Drilling
Engine Idling
Predicted class



6) After playing around the parameters/hyperparameters, the model below is the best. It has reasonable training time and prediction speed.

Last change: 'Cost matrix' = 'custom' 13/13 features

11 ☆ Ensemble Accuracy: 54.9%

Last change: 'Number of learners' = '100' 13/13 features

▾ Current Model

Model 11: Trained

Results

Accuracy	54.9%
Total misclassification cost	1471 (Models have different cost matrices)
Prediction speed	~1700 obs/sec
Training time	28.244 sec

Model Type

Preset: Bagged Trees
Ensemble method: Bag
Learner type: Decision tree
Maximum number of splits: 2619
Number of learners: 100

Optimizer Options

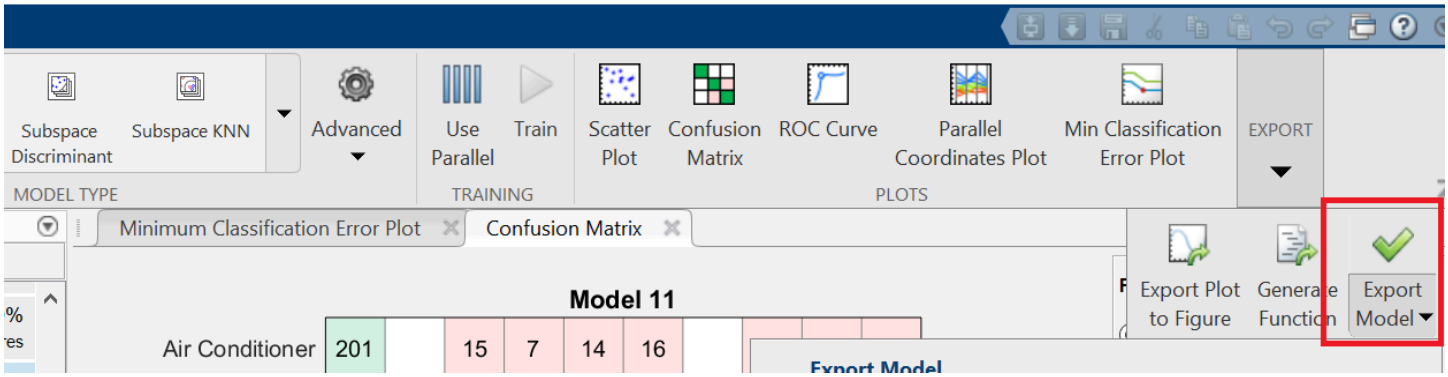
Hyperparameter options disabled

Feature Selection

All features used in the model, before PCA

Since i could not able to increase the accuracy significantly, as expected from visualizing the data before machine learning, we can't get model with good accuracy.

7) Export the model

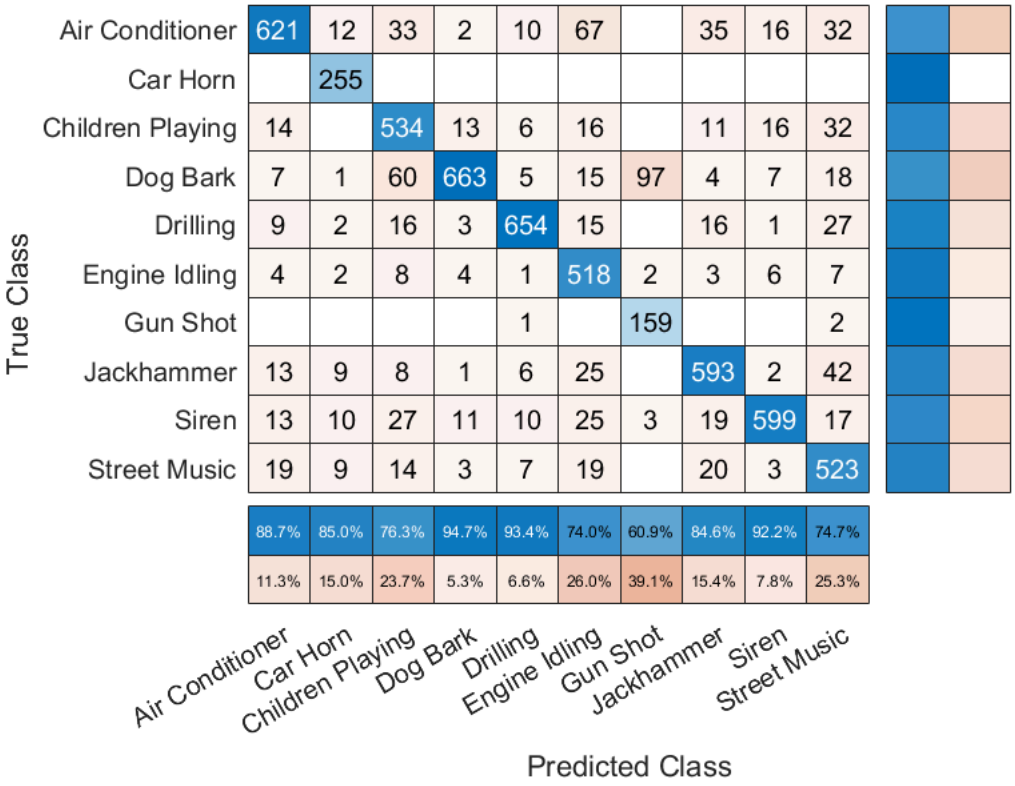


Export Classification Model to Predict New Data

Make Predictions for New Data

After you export a model to the workspace from Classification Learner, or run the code generated from the app, you get a `trainedModel` structure that you can use to make predictions using new data. The structure contains a classification object and a function for prediction

```
[predictionoutcome,scores]=trainedModel.predictFcn(dataTest);
fig = figure;
cm = confusionchart(string(predictionoutcome),dataTest.Label,'RowSummary','row-normalized','Col
```




```
accuracy=sum(string(predictionoutcome)==dataTest.Label)/length(dataTest.Label)
```

```
accuracy = 0.8375
```

Accuracy is 83.75%

Observation :

We have noted down some important information before training, let review again :

Observation from ML1

1) With our eye, we can differenciate the sound using their shape of signal. Am I right? Now, we take note about it first.

- Shape factor is the most important factor in our ranking, therefore, we prove it right. However, with time domain and spectral features, we dont have much information about the shape, hence, we only able to hit the prediction up to 55%.

Observation from ML2

1) if we look at the shape of signal, we will notice some are looking similar to each other although we still can differentiate it through sound. Especially, gun shot is looking smilar to jackhammer. Drilling and Engine Idling are also looking similar.

- It is correct. In my intuitive, i try to impose higher penalty for the cost matrix to deal with this, however, I failed to increase the accuracy significantly.

2) if we are looking at the graph, many classes does not have significant distinction, we might dont have good accuracy result if we only use time-domain and spectral features to train machine learning model.

- Now it is time for us to think what other features we can included which the features will tell more information about the shape. MFCC or Wavelet transform?

Generate MATLAB Code to Train the Model with New Data

After you create classification models interactively in Classification Learner, you can generate MATLAB code for your best model. You can then use the code to train the model with new data.

```
saveLearnerForCoder(trainedModel.ClassificationEnsemble,'bagtree');
```

```
ans = 1×1 cell array
    {'Air Conditioner'}
```

```
function label = classifyurbansound(X) %#codegen
Mdl = loadLearnerForCoder('bagtree');
label = predict(Mdl,X);
end
```