



# EECE5644 Machine Learning Final Project **Urban Sound Classification**

Zachary Neveu, Tianyi Zhou, Christian Grenier

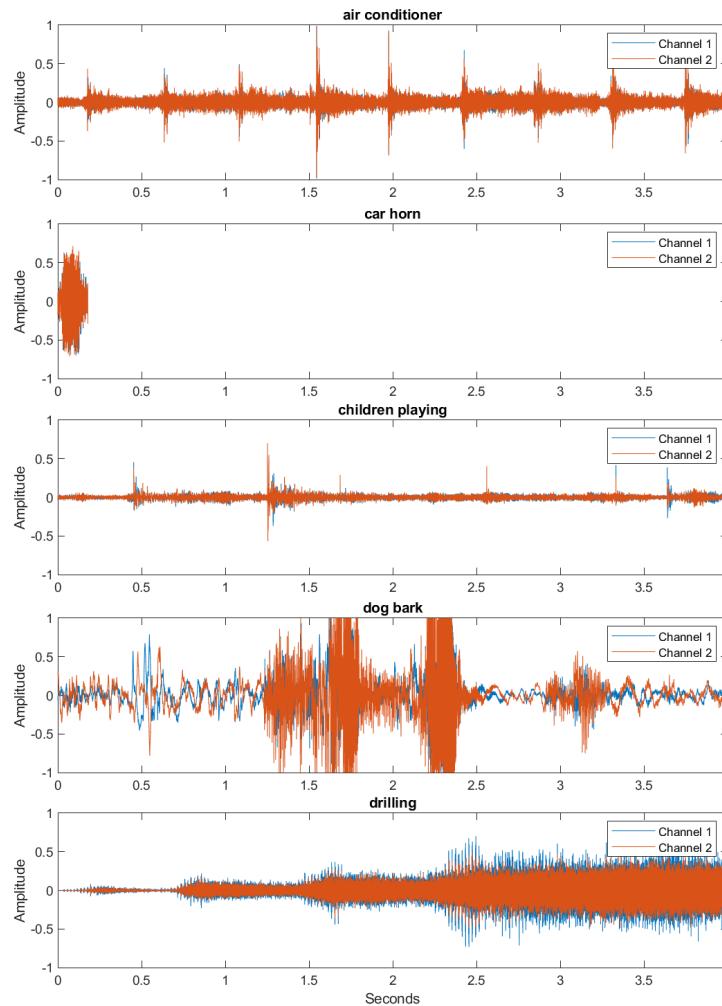
Northeastern University

04-16-2019

# Task: Classify Urban Sounds into 10 Categories

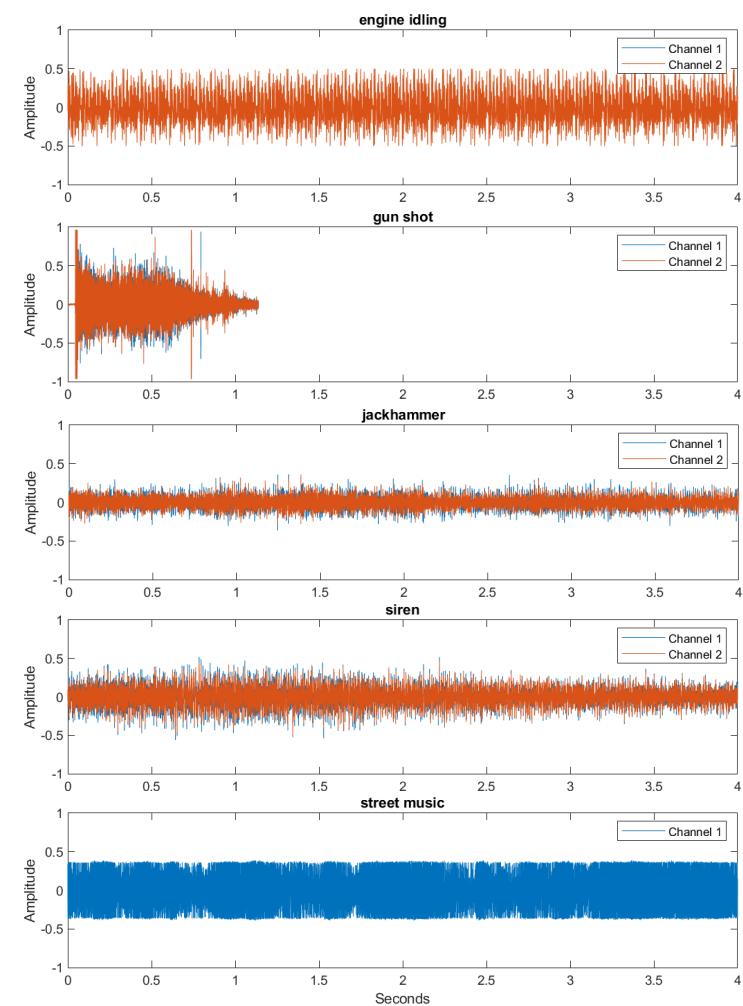
## Dataset

- [UrbanSound8k](#) from Kaggle
- **5435** Audio .wav Files with 10 classes (shown below)
- Duration: **0.05 ~ 4s**
- Sample Rates: **44.1 kHz, 48 kHz, 96 kHz**



## Potential Uses

- Gun violence detection
- Predicting travel times based on construction
- Predicting energy usage based on air conditioner levels
- Acoustic situational awareness for machines





# Pre-Processing

- [UrbanSound8k](#) Dataset from Kaggle is split:
  - Training Set: **5435** labeled audio files → our data set
  - Test Set: **3297** unlabeled audio files → only for competition on Kaggle
- The **audiodatastore** object is used to manage the collection of audio files, and the **audioread** function extracts each file from the object.
- Files are sorted in numerical order instead of alphabetical,... as shown below. This way the labels match

dataStore		dataStore.Files
dataStore.Files		
1		
1	G:\My Drive\EECE5644_MachineLearning\Final_Project\RAWDATAUnzipped\train\Train\0.wav	
2	G:\My Drive\EECE5644_MachineLearning\Final_Project\RAWDATAUnzipped\train\Train\1.wav	
3	G:\My Drive\EECE5644_MachineLearning\Final_Project\RAWDATAUnzipped\train\Train\10.wav	
4	G:\My Drive\EECE5644_MachineLearning\Final_Project\RAWDATAUnzipped\train\Train\100.wav	
5	G:\My Drive\EECE5644_MachineLearning\Final_Project\RAWDATAUnzipped\train\Train\1000.wav	
6	G:\My Drive\EECE5644_MachineLearning\Final_Project\RAWDATAUnzipped\train\Train\1001.wav	
7	G:\My Drive\EECE5644_MachineLearning\Final_Project\RAWDATAUnzipped\train\Train\1003.wav	
8	G:\My Drive\EECE5644_MachineLearning\Final_Project\RAWDATAUnzipped\train\Train\1004.wav	
9	G:\My Drive\EECE5644_MachineLearning\Final_Project\RAWDATAUnzipped\train\Train\1006.wav	
10	G:\My Drive\EECE5644_MachineLearning\Final_Project\RAWDATAUnzipped\train\Train\1007.wav	
11	G:\My Drive\EECE5644_MachineLearning\Final_Project\RAWDATAUnzipped\train\Train\1008.wav	
12	G:\My Drive\EECE5644_MachineLearning\Final_Project\RAWDATAUnzipped\train\Train\101.wav	
13	G:\My Drive\EECE5644_MachineLearning\Final_Project\RAWDATAUnzipped\train\Train\1014.wav	
14	G:\My Drive\EECE5644_MachineLearning\Final_Project\RAWDATAUnzipped\train\Train\1015.wav	
15	G:\My Drive\EECE5644_MachineLearning\Final_Project\RAWDATAUnzipped\train\Train\1017.wav	
16	G:\My Drive\EECE5644_MachineLearning\Final_Project\RAWDATAUnzipped\train\Train\1018.wav	
17	G:\My Drive\EECE5644_MachineLearning\Final_Project\RAWDATAUnzipped\train\Train\1021.wav	
18	G:\My Drive\EECE5644_MachineLearning\Final_Project\RAWDATAUnzipped\train\Train\1022.wav	
19	G:\My Drive\EECE5644_MachineLearning\Final_Project\RAWDATAUnzipped\train\Train\1024.wav	
20	G:\My Drive\EECE5644_MachineLearning\Final_Project\RAWDATAUnzipped\train\Train\1025.wav	

	\in\Train\0.wav
	\in\Train\1.wav
	\in\Train\2.wav
	\in\Train\3.wav
	\in\Train\4.wav
	\in\Train\6.wav
	\in\Train\10.wav
	\in\Train\11.wav
	\in\Train\12.wav
	\in\Train\15.wav
	\in\Train\17.wav
	\in\Train\18.wav
	\in\Train\19.wav
	\in\Train\20.wav
	\in\Train\22.wav
	\in\Train\24.wav
	\in\Train\26.wav
	\in\Train\27.wav
	\in\Train\32.wav
	\in\Train\33.wav
	\in\Train\35.wav

- Second channel of audio data is ignored for simplification and uniformity

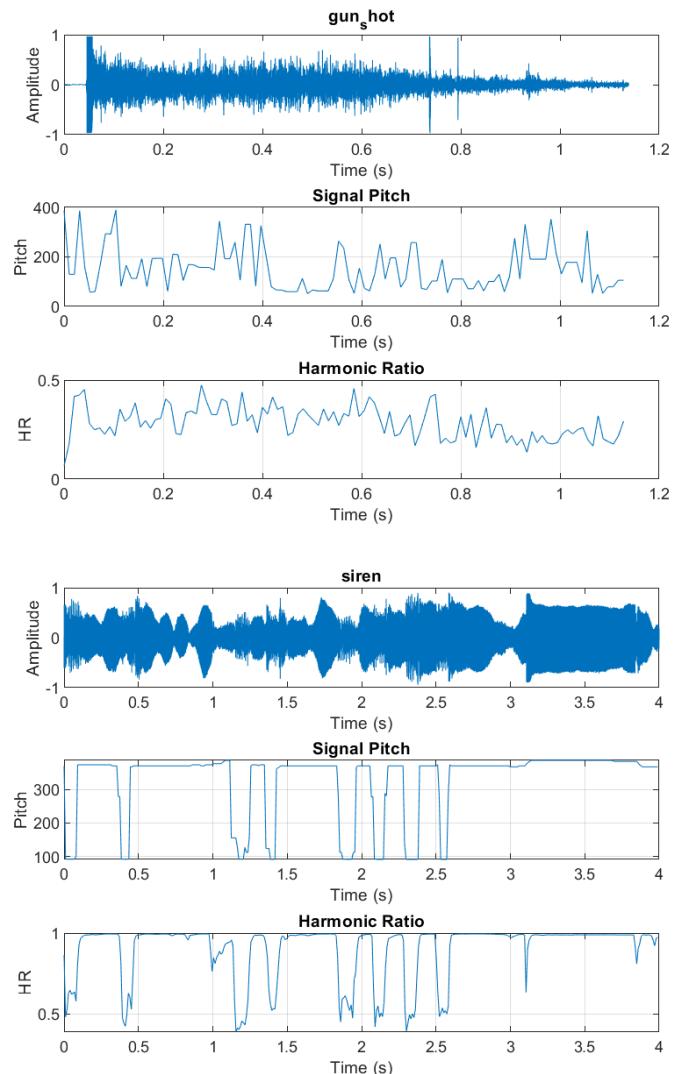
# Feature Extraction

- **Basic Info (# of features: 2)**
  - Sample Rate
  - Duration
- **Loudness (2)**
  - Integrated Loudness
  - Loudness Range
    - Max loudness – Min loudness
- **Pitch (2)**
  - Average Pitch
  - Average Magnitude of Derivative of Pitch
- **Harmonic Ratio (3)**
  - Average HR
  - Variance of HR
  - Average Magnitude Derivative of HR
- **Mel Frequency Cepstral Coefficients (98)**
  - 7 time windows for each audio signal
  - 1 log energy value and 13 Mel frequency cepstral coefficients (MFCC) for each time window
  - See more details about MFCC in next slide

## NOTE:

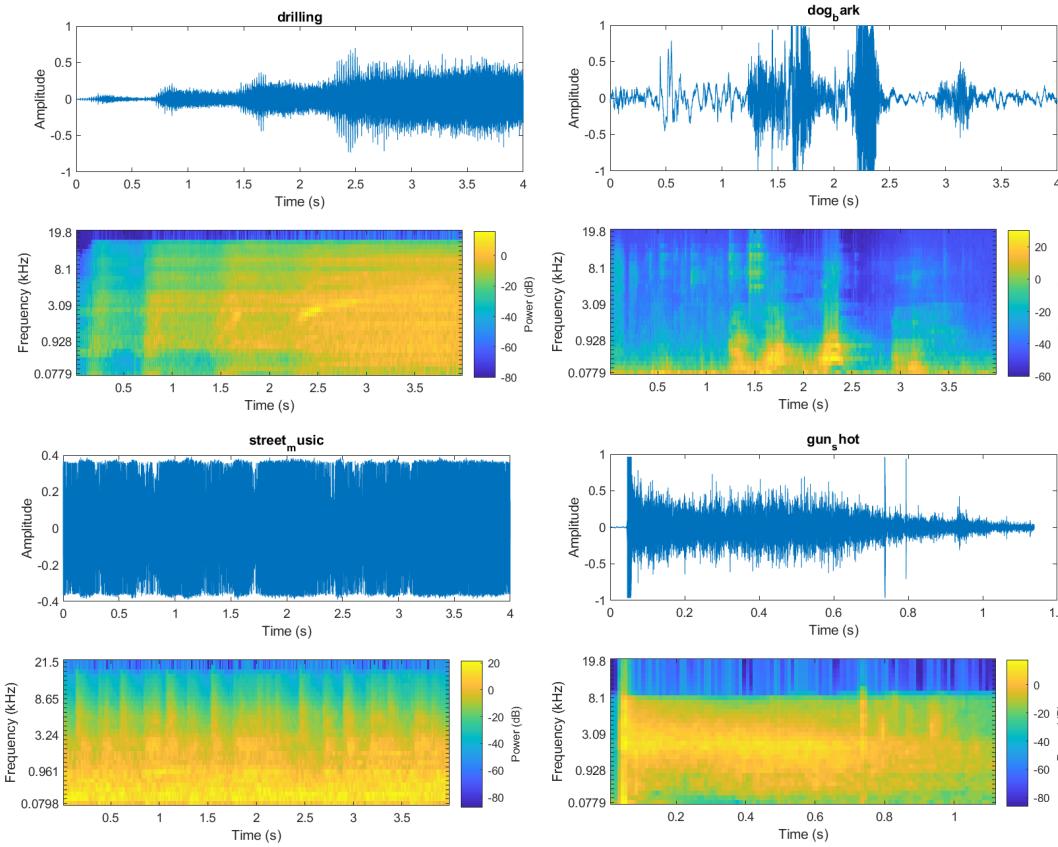
1. Audio Toolbox in MATLAB provides: `integratedLoudness`, `pitch`, `harmonicRatio`, `mfcc` functions to extract features from audios.
2. We padded the short audio to 4s, and added small SNR gaussian white noise so that the mfcc function doesn't return  $-\infty$  when taking the log of the DFT. Then we manually set the window length and overlap length so that the number of window is 7.

Example plots  
of Pitch and HR  
for gunshot and  
siren audio.



# Mel Frequency Cepstral Coefficients

**Mel Frequency Cepstrum (MFC)** is a representation of the short-term power spectrum of the audio. It is derived by dividing the signal into time windows, taking Fourier Transform within each window, then mapping the powers of the spectrum onto the mel scale. MFC for selected audio samples are shown below.



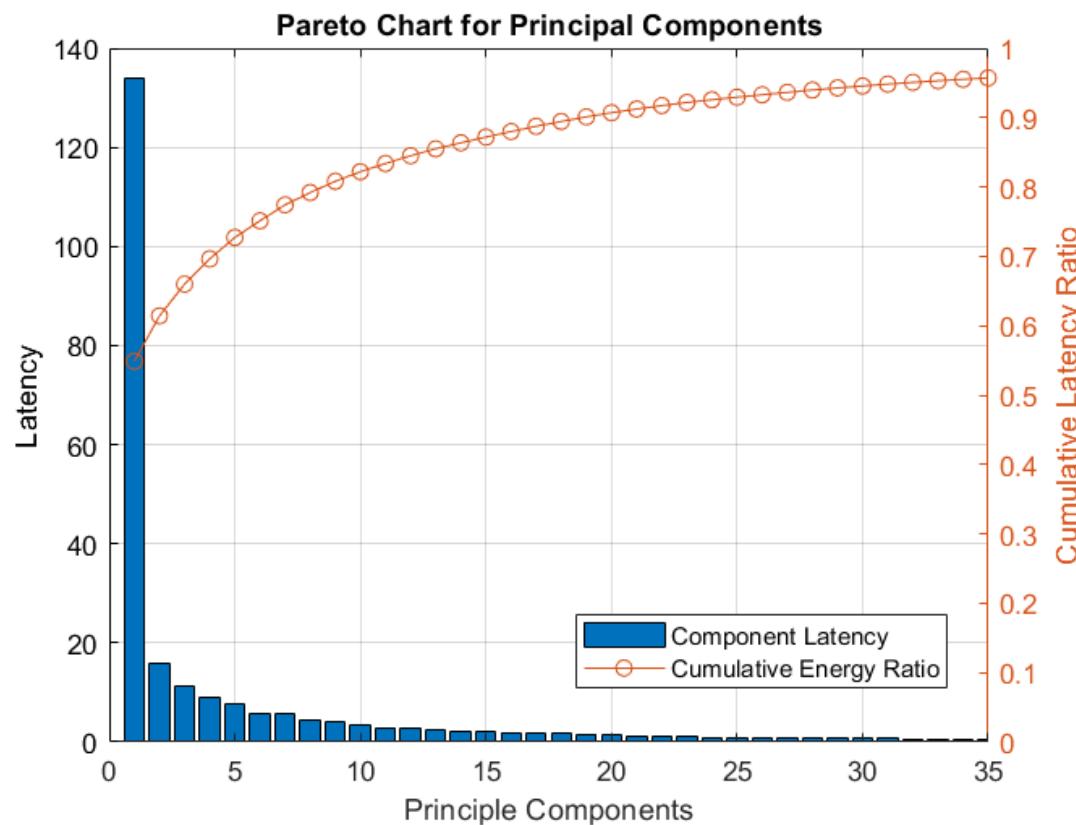
- Mel Frequency Cepstral Coefficients (MFCCs) are coefficients that collectively make up an MFC.
- As shown, **the spectrograms accurately capture time and frequency data for each audio class.**

To classify an audio sample, we could either:

1. Take MFCCs as features and perform **Supervised Classification**
2. Take MFCs to do **Image Classification using Convolutional Neural Network**

# Dimensionality Reduction - PCA

- A customized zscore function is used to deal with missing data in feature array
- Principal Component Analysis is performed with a threshold of 95% cumulative latency ratio
  - **107 → 32 features**





# Supervised Classification

## Model Selection

- The [Classification Learner](#) in MATLAB allows us to explore various classifiers and search for the best classification model type.
- We select 5 models with the best performances, manually tune the parameters for the classification model. These models are:
  - Ensemble Subspace KNN
  - K-Nearest Neighbor
  - Cubic Kernel SVM
  - Quadratic Kernel SVM
  - Gaussian Kernel SVM

Command Window

New to MATLAB? See resources for [Getting Started](#).

```
>> clear
>> close all

.....Starting PCA.....
Elapsed time is 25.947758 seconds.

.....Training classification models for 10-fold CV.....
Elapsed time is 4.964392 seconds.

.....Predicting and plotting confusion matrices.....
Accuracy: 87.4%. CV: 10-fold. Model: Subspace KNN Ensemble.
Accuracy: 85.3%. CV: 10-fold. Model: KNN.
Accuracy: 84.5%. CV: 10-fold. Model: Cubic SVM.
Accuracy: 84.2%. CV: 10-fold. Model: Quadratic SVM.
Accuracy: 83.6%. CV: 10-fold. Model: Gaussian SVM.
Elapsed time is 47.660765 seconds.

.....Training classification models for 80-20 Holdout CV.....
Elapsed time is 1.124702 seconds.

.....Predicting and plotting confusion matrices.....
Accuracy: 87.0%. CV: 80-20 Holdout. Model: Subspace KNN Ensemble.
Accuracy: 85.0%. CV: 80-20 Holdout. Model: KNN.
Accuracy: 84.8%. CV: 80-20 Holdout. Model: Cubic SVM.
Accuracy: 84.2%. CV: 80-20 Holdout. Model: Quadratic SVM.
Accuracy: 83.7%. CV: 80-20 Holdout. Model: Gaussian SVM.
Elapsed time is 3.515305 seconds.
```

f> >> |

1.1 ★ Linear Discriminant Last change: Linear Discriminant	Accuracy: 53.2% 32/32 features
1.2 ★ Quadratic Discriminant Last change: Quadratic Discriminant	Accuracy: 65.8% 32/32 features
2 ★ SVM Last change: Linear SVM	Accuracy: 66.2% 32/32 features
3 ★ SVM Last change: Quadratic SVM	Accuracy: 83.1% 32/32 features
4 ★ SVM Last change: Cubic SVM	Accuracy: 84.6% 32/32 features
5 ★ SVM Last change: Medium Gaussian SVM	Accuracy: 82.6% 32/32 features
6 ★ KNN Last change: Fine KNN	Accuracy: 82.7% 32/32 features
7 ★ Ensemble Last change: Subspace KNN	Accuracy: 88.3% 32/32 features
8 ★ Ensemble Last change: Subspace Discriminant	Accuracy: 52.4% 32/32 features

# Supervised Classification

## Cross Validation and Model Evaluation

- Both K-fold (with 10 folds) and Holdout (80-20 split) methods of CV are used
- Parallel computing improves the speed of training and validation for 80-20 holdout case.
- Selected confusion matrices are shown below.

True Class

Subspace KNN Ensemble Confusion Matrix 10-fold CV, Acc:87.4%										
	air_conditioner	car_horn	children_playing	dog_bark	drilling	engine_idling	gun_shot	jackhammer	siren	street_music
air_conditioner	583	4	2	1			5	1	4	
car_horn	1	243	2	19	9	3	13	9		7
children_playing	21	2	480	15	8	2	1	9	5	57
dog_bark	6	33	65	406	24	2	18	8	14	24
drilling	1	4	4	6	537	2	21	21		4
engine_idling	6	1	2	2	2	597	4	5	2	3
gun_shot		5	2	6	8		204	5		
jackhammer	1			2	7		11	643		4
siren	5	1	17	6		2	1	1	566	8
street_music	33	5	30	6	4	5		13	14	490

Predicted Class

True Class

Subspace KNN Ensemble Confusion Matrix 80-20 Holdout CV, Acc:87%										
	air_conditioner	car_horn	children_playing	dog_bark	drilling	engine_idling	gun_shot	jackhammer	siren	street_music
air_conditioner	119								1	
car_horn		49		3	1		6			2
children_playing	8		94	3				5		10
dog_bark		2	14	88	5		2	3	2	5
drilling			1	108	1	4	4			2
engine_idling	2		3		1	113	4			1
gun_shot		2	1		2		41			
jackhammer				3		1	129			
siren	1	2	1						116	1
street_music	8	3	11	5			2	4	88	

Predicted Class

True Class

KNN Confusion Matrix 10-fold CV, Acc:85.3%										
	air_conditioner	car_horn	children_playing	dog_bark	drilling	engine_idling	gun_shot	jackhammer	siren	street_music
air_conditioner	587	4	2		1	1	3		2	
car_horn	1	228		25	16	5	16	8	1	6
children_playing	24	12	450	17	7	11	2	7	9	61
dog_bark	8	43	43	381	29	12	37	10	11	26
drilling	5	9	1	2	519	4	32	22	1	5
engine_idling	2	4	4		3	596	6	1	3	5
gun_shot	1	8	2	8	16	2	183	10		
jackhammer		2	1	3	9	1	18	631		3
siren	6	1	8	8		2	1	1	569	11
street_music	24	8	24	16	6	10		15	7	490

Predicted Class

True Class

KNN Confusion Matrix 80-20 Holdout CV, Acc:85%										
	air_conditioner	car_horn	children_playing	dog_bark	drilling	engine_idling	gun_shot	jackhammer	siren	street_music
air_conditioner	117		1	1					1	
car_horn		46		6	3			5		1
children_playing	7	2	84	3		2		4	3	15
dog_bark		3	8	89	5	3	5	3	1	4
drilling	2		1	102	1	6	6	1	1	
engine_idling			1		2	117	1	2		1
gun_shot	2	1	2	3		35	3			
jackhammer		3			3	1		126		
siren			2	2		1			115	1
street_music	7	1	7	3		5		1	5	92

Predicted Class

True Class

Cubic SVM Confusion Matrix 10-fold CV, Acc:84.5%											
	air_conditioner	car_horn	children_playing	dog_bark	drilling	engine_idling	gun_shot	jackhammer	siren	street_music	
air_conditioner	566		1	1				3	11	3	15
car_horn		233	3	15	15	8	9	8		15	
children_playing	11	1	438	36	17	5		6	12	74	
dog_bark	7	18	46	430	26	10	10	5	17	31	
drilling	2	7	4	10	528	12	8	19		10	
engine_idling	13	1	3	3	2	577	4	3	6	12	
gun_shot	1	2	2	4	7	6	206				
jackhammer	5	4	3	1	11	3	4	631		6	
siren	6	2	32	12	3	5		1	528	18	
street_music	8	11	62	14	11	11		7	21	455	

Predicted Class

True Class

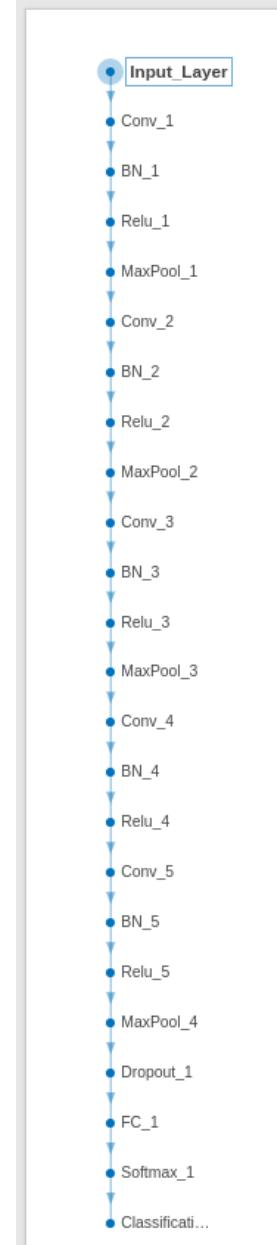
Cubic SVM Confusion Matrix 80-20 Holdout CV, Acc:84.6%										
	air_conditioner	car_horn	children_playing	dog_bark	drilling	engine_idling	gun_shot	jackhammer	siren	street_music
air_conditioner	113						1	2	4	
car_horn		45	1	4	2	2	1	2		4
children_playing	3		92	5	1	1		2	1	15
dog_bark	1	2	4	101	7	2				2
drilling	1	1	2	3	101	4	1	4		3
engine_idling	3		3		1	109	3	2	2	1
gun_shot		2	1	1		1	41			
jackhammer	2	1			2			126		2
siren	1	3	5	2					105	5
street_music	3	3	13	5	5	2		1	3	86

Predicted Class



# Neural Network Architecture

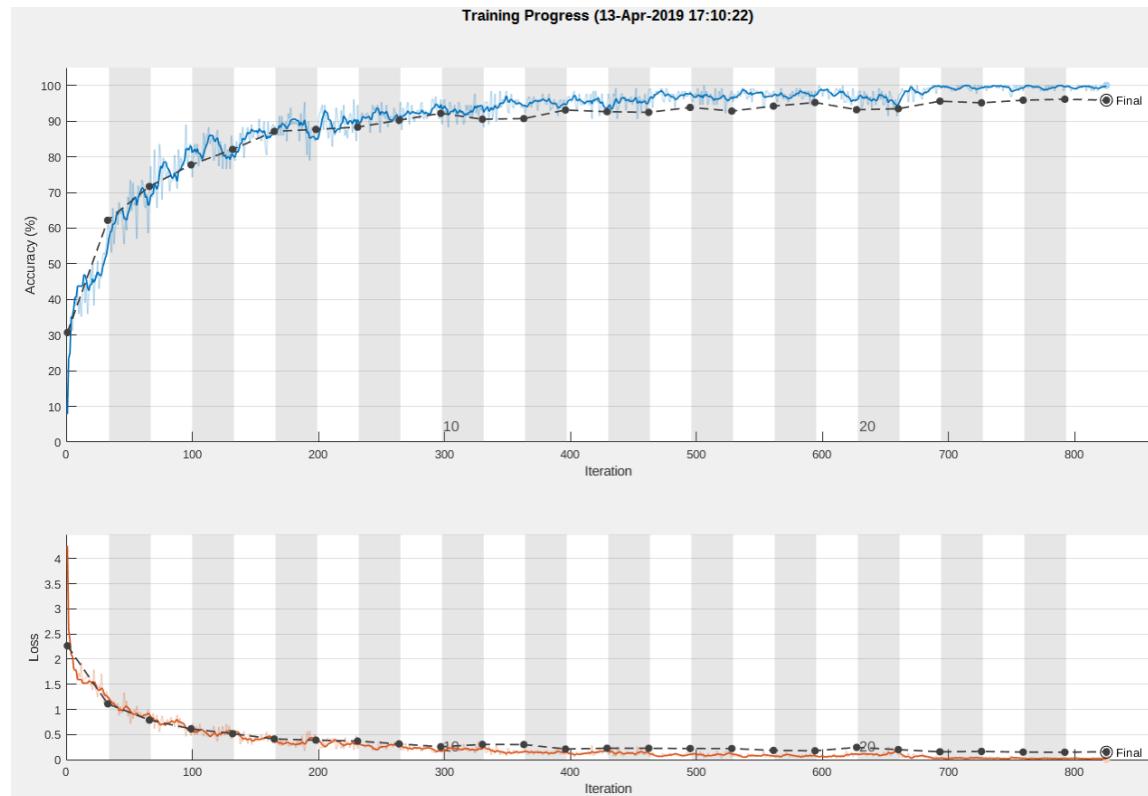
- Convolutional
- Repeated Blocks of:
  - 3x3 Convolution
  - Batch Normalization
  - RELU
  - Max Pool
- Dropout layer to avoid overfitting
- Softmax layer to create one-class predictions
- 24 Layers Deep



ANALYSIS RESULT			
	Name	Type	Activations
1	Input_Layer 40x396x1 images with 'zerocenter' normalization	Image Input	40x396x1
2	Conv_1 12 3x3x1 convolutions with stride [1 1] and padding 'same'	Convolution	40x396x12
3	BN_1 Batch normalization with 12 channels	Batch Normalization	40x396x12
4	Relu_1 ReLU	ReLU	40x396x12
5	MaxPool_1 3x3 max pooling with stride [2 2] and padding 'same'	Max Pooling	20x198x12
6	Conv_2 24 3x3x12 convolutions with stride [1 1] and padding 'same'	Convolution	20x198x24
7	BN_2 Batch normalization with 24 channels	Batch Normalization	20x198x24
8	Relu_2 ReLU	ReLU	20x198x24
9	MaxPool_2 3x3 max pooling with stride [2 2] and padding 'same'	Max Pooling	10x99x24
10	Conv_3 48 3x3x24 convolutions with stride [1 1] and padding 'same'	Convolution	10x99x48
11	BN_3 Batch normalization with 48 channels	Batch Normalization	10x99x48
12	Relu_3 ReLU	ReLU	10x99x48
13	MaxPool_3 3x3 max pooling with stride [2 2] and padding 'same'	Max Pooling	5x50x48
14	Conv_4 48 3x3x48 convolutions with stride [1 1] and padding 'same'	Convolution	5x50x48
15	BN_4 Batch normalization with 48 channels	Batch Normalization	5x50x48
16	Relu_4 ReLU	ReLU	5x50x48
17	Conv_5 48 3x3x48 convolutions with stride [1 1] and padding 'same'	Convolution	5x50x48
18	BN_5 Batch normalization with 48 channels	Batch Normalization	5x50x48
19	Relu_5 ReLU	ReLU	5x50x48
20	MaxPool_4 1x50 max pooling with stride [1 1] and padding [0 0 0 0]	Max Pooling	5x1x48
21	Dropout_1 20% dropout	Dropout	5x1x48
22	FC_1 10 fully connected layer	Fully Connected	1x1x10
23	Softmax_1 softmax	Softmax	1x1x10
24	Classification crossentropyex with 'air_conditioner' and 9 other classes	Classification Output	-

# Neural Network Training

- 80%/20% training/validation split in data
- ADAM Optimizer (see <https://arxiv.org/abs/1412.6980>)
  - Faster than SGD with momentum
- 25 Epochs to converge
- Maximum Validation Accuracy: 95.86%



# Results and Conclusion

- The best classification model is Ensemble Subspace KNN, with 10-fold CV accuracy of **~87%**.
- To improve results:
  - Include second channel of audio files
  - Increase frequency and time resolution for MFCCs
  - Better pre-processing that includes pitch values for short duration files
- A Convolutional Neural Network achieves a maximum accuracy of **95.86%** on a 20% hold out validation set

Subspace KNN Ensemble Confusion Matrix 10-fold CV, Acc:87.4%											
True Class	air_conditioner	583		4	2	1			5	1	4
	car_horn	1	243	2	19	9	3	13	9		7
	children_playing	21	2	480	15	8	2	1	9	5	57
	dog_bark	6	33	65	406	24	2	18	8	14	24
	drilling	1	4	4	6	537	2	21	21		4
	engine_idling	6	1	2	2	2	597	4	5	2	3
	gun_shot		5	2	6	8		204	5		
	jackhammer	1			2	7		11	643		4
	siren	5	1	17	6		2	1	1	566	8
	street_music	33	5	30	6	4	5		13	14	490
Predicted Class											
air_conditioner	car_horn	children_playing	dog_bark	drilling	engine_idling	gun_shot	jackhammer	siren	street_music		

Neural Network Confusion Matrix on 20% Validation Set										
True Class	air_conditioner	117	1	1						1
	car_horn		60		2	1				2
	children_playing			116	9		1			11
	dog_bark	1			103					1
	drilling				4	118	1	1		
	engine_idling	1				122				1
	gun_shot				1		45			
	jackhammer					1		134		2
	siren	1			1				118	
	street_music		3		1					105
Predicted Class										
air_conditioner	car_horn	children_playing	dog_bark	drilling	engine_idling	gun_shot	jackhammer	siren	street_music	