

Audio Classification

<https://github.com/ItaiShchorry/AudioClassifier>

Itai Shchorry

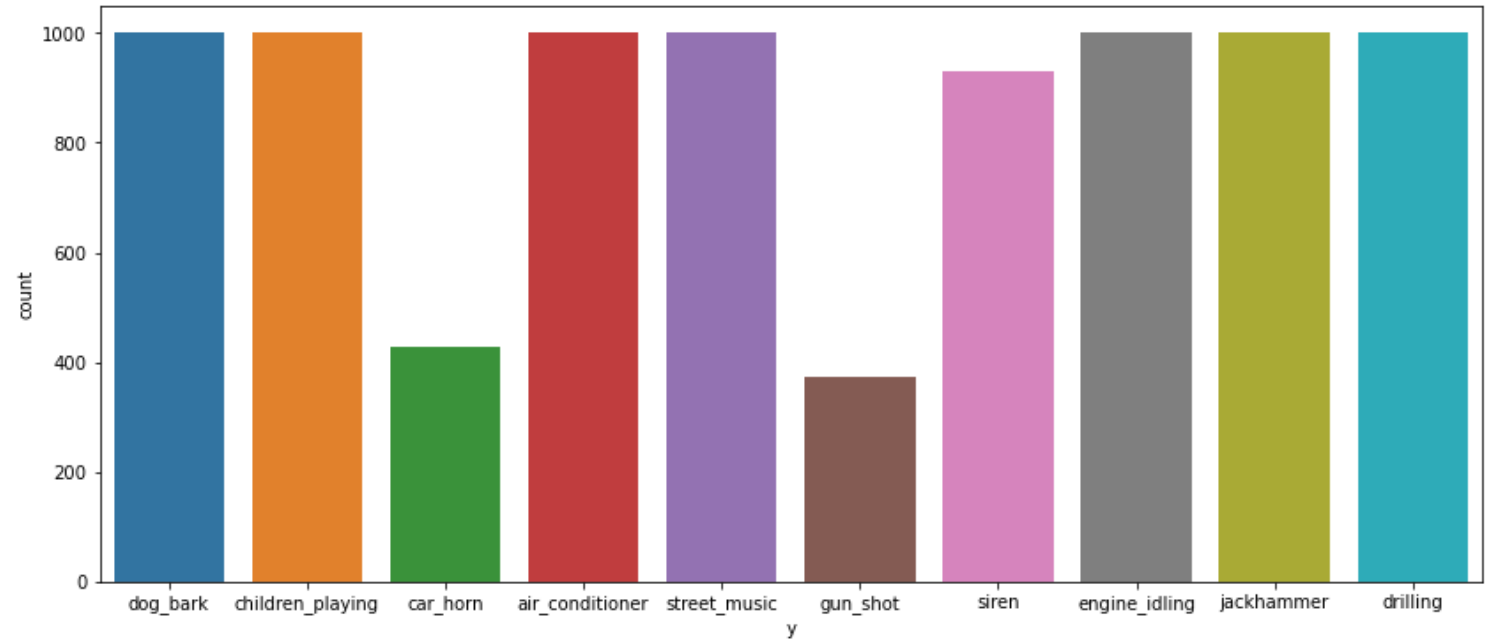
Overview

- Project Goal – build a template methodology for developing & deploying ML applications
 - Modular & reusable as possible
- Approach – Build an audio classifying tool
 - Assumption – audio can be viewed as tabular, sequential and window-based, hence developing a utility for audio potentially extends to supporting all these datatypes

Dataset

- Urban Sounds Dataset

- 10 classes
- 9K examples



Baselines

Classical Tabular Models

	Model	Description	Data Version	Accuracy	Kappa	Run Time
1	Baseline Classification	Random Baseline	Edition 1	0.117939	0.011735	0:00:00
2	Decision Tree	Default settings	Edition 1	0.672901	0.633752	0:00:00
3	Random Forest 1	Default settings	Edition 1	0.891221	0.877943	0:00:00
4	Random Forest 2	Greedy Feature Selection, n_estimators=40, max_features = 1	Edition 1	0.861832	0.844998	0:00:00
5	Random Forest 3	Very Shallow, min_samples_leaf=0.1	Edition 1	0.612595	0.564400	0:00:00
6	Random Forest 4	Greedy Feature Selection, min_samples_leaf=0.01, max_features=1	Edition 1	0.612595	0.564400	0:00:00
7	Random Forest 5	Less Shallow, Greedy, min_samples_leaf=0.001, max_features=0.8	Edition 1	0.810687	0.787507	0:00:00
8	Random Forest 6	Less Greedy FS, min_samples_leaf=0.001, max_features=0.8	Edition 1	0.811832	0.788772	0:00:00
9	Random Forest 7	Regularized, min_samples_leaf=0.001, max_features=sqrt	Edition 1	0.828244	0.807191	0:00:00
10	Gradient Boosting	Default settings	Edition 1	0.815267	0.792760	0:01:16
11	XGBoost	Default settings	Edition 1	0.891221	0.877997	0:00:30
12	LightGBM	Default settings	Edition 1	0.901527	0.889529	0:00:07

Fast.ai's Tabular Learner (FC-based NN)

Overfit a single batch

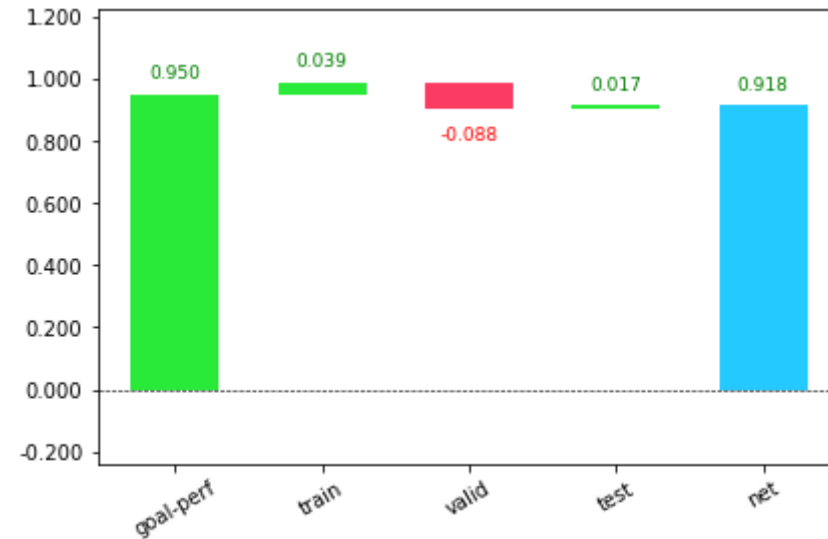
Train with defaults. Score – 90.07% (accuracy)

Bias-variance decomp. In next slide

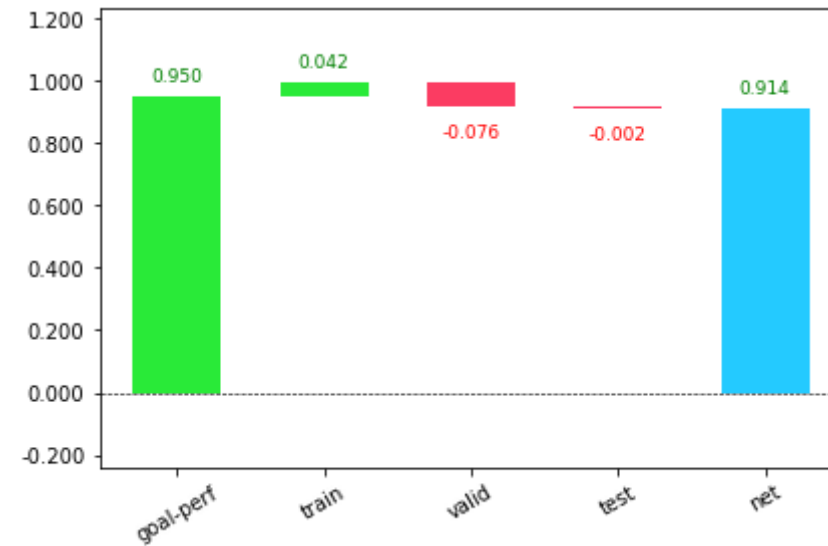
Confusion matrix

Actual \ Predicted	air_conditioner	car_horn	children_playing	dog_bark	drilling	engine_idling	gun_shot	jackhammer	siren	street_music
air_conditioner	148	0	0	0	0	1	0	0	0	1
car_horn	0	58	0	1	1	2	0	0	0	2
children_playing	1	0	135	5	0	1	2	0	0	6
dog_bark	2	2	16	116	2	1	2	0	4	5
drilling	0	0	0	3	140	0	0	2	0	5
engine_idling	2	0	4	0	0	137	0	0	3	4
gun_shot	0	0	2	3	1	0	49	0	0	1
jackhammer	0	0	0	0	3	1	0	144	0	2
siren	2	0	1	2	1	0	0	0	133	1
street_music	1	0	12	5	2	2	1	2	5	120

Bias-Variance Decomposition



Observing Overfitting -> Adding regularizations (weight decay)



Less overfitting but worse test accuracy. We'll try an HP search.

Experiment Tracking & Hyperparameter Optimization

- Experiment Tracking
 - W&B
- HP Optimization
 - Data aspects as hyperparameters (outlier-dropping thresholds)
 - Grid-search based (focus on lr, layer groups momentums & weight decay)

<input type="checkbox"/> Name (144 visualized)	Runtime	Notes	Tags	Learner.lr	Learner.mo	Learner.wd	accuracy ▾	lr_0	mom_0	raw_loss	train_loss	valid_loss	wd_0
<input type="checkbox"/> tuning_cutoff_0_16	4m 24s	Add notes		0.005	[0.85,0.9,0	0.1	0.9443	5.152e-8	0.95	0.01312	0.01518	0.2361	0.1
• tuning_cutoff_0_22	4m 28s	Add notes		0.005	[0.9,0.9,0.!	0.1	0.9427	5.152e-8	0.9	0.03256	0.01595	0.2344	0.1
• tuning_cutoff_0_20	4m 39s	Add notes		0.005	[0.95,0.9,0	0.01	0.9427	5.152e-8	0.85	0.01897	0.01454	0.2747	0.01
• tuning_cutoff_0_13	4m 54s	Add notes		0.005	[0.95,0.85,	0.1	0.9412	5.152e-8	0.95	0.01395	0.01623	0.2311	0.1
• tuning_cutoff_0_21	4m 45s	Add notes		0.005	[0.9,0.9,0.!	-	0.9397	5.152e-8	0.9	0.004743	0.01195	0.2679	0.01
• tuning_cutoff_0_12	4m 55s	Add notes		0.005	[0.95,0.85,	-	0.9397	5.152e-8	0.95	0.009253	0.01102	0.2999	0.01
• tuning_cutoff_0_15	4m 22s	Add notes		0.005	[0.85,0.9,0	-	0.9397	5.152e-8	0.95	0.003691	0.01325	0.2466	0.01
• tuning_cutoff_0_17	4m 22s	Add notes		0.005	[0.85,0.9,0	0.01	0.9389	5.152e-8	0.95	0.01403	0.0165	0.3068	0.01
• tuning_cutoff_0_19	4m 58s	Add notes		0.005	[0.95,0.9,0	0.1	0.9389	5.152e-8	0.85	0.02006	0.01769	0.2346	0.1
• tuning_cutoff_0_18	4m 37s	Add notes		0.005	[0.95,0.9,0	-	0.9374	5.152e-8	0.85	0.02184	0.01274	0.3014	0.01
• tuning_cutoff_0_23	4m 38s	Add notes		0.005	[0.9,0.9,0.!	0.01	0.9366	5.152e-8	0.9	0.03111	0.01313	0.3094	0.01
• tuning_cutoff_5_20	4m 16s	Add notes		0.005	[0.95,0.9,0	0.01	0.9359	5.152e-8	0.85	0.006952	0.01182	0.3244	0.01
• tuning_cutoff_5_19	4m 29s	Add notes		0.005	[0.95,0.9,0	0.1	0.9351	5.152e-8	0.85	0.01159	0.01572	0.2809	0.1
• tuning_cutoff_5_16	4m 29s	Add notes		0.005	[0.85,0.9,0	0.1	0.9351	5.152e-8	0.95	0.01948	0.01507	0.2584	0.1

Managed to produce ~3% improvement

Deployment

- Based on Streamlit
- Supports EDA, Evaluations Drilldown & Inferencing

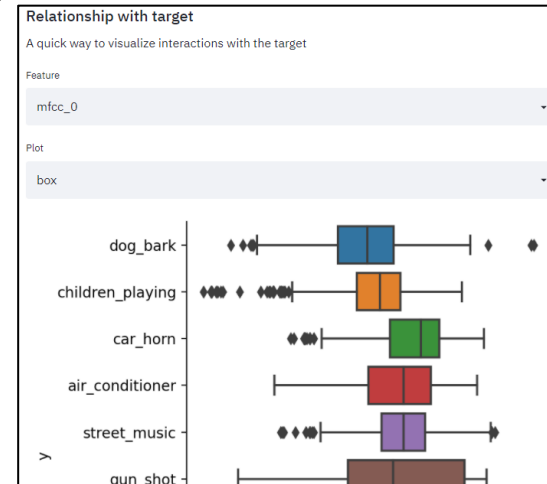
Navigation

Go to

- ☒ Data Explorer
- ☐ Model Evaluator
- ☐ Inferencer

Data Explorer

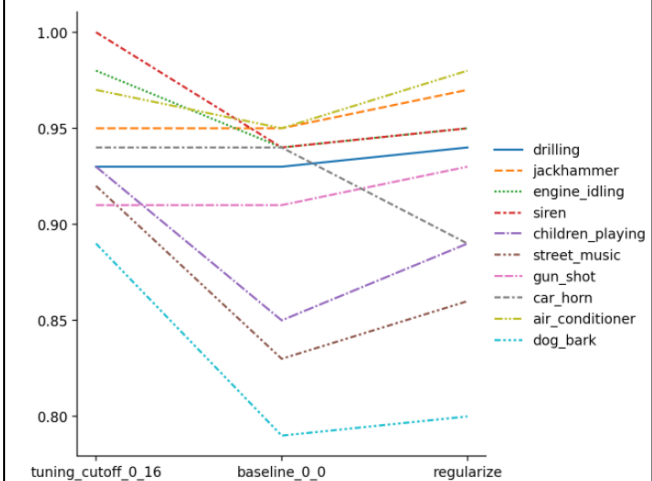
- ☒ Data Summary
- ☐ Target Exploration
- ☐ Correlations



General Performance

	tr_acc	val_acc	test_acc
tuning_cutoff_0_16	0.9985	0.9443	0.9252
baseline_0_0	0.9910	0.9800	0.9892
regularize	0.9921	0.9160	0.9137

Class-level Performance



Model Drilldown

Choosing population

Which model would you like to predict with?

tuning_cutoff_0_16

Inference

upload a wave file for prediction

Drag and drop file here
Limit 200MB per file

Browse files

7061-6-0-0.wav 387.6KB

You uploaded this sound

0:00 / 0:02

And the classification probabilities using tuning_cutoff_0_16 are:

	class	probability
6	gun_shot	0.9940
4	drilling	0.0054
3	dog_bark	0.0005
8	siren	0.0001
7	jackhammer	0.0000

Next Steps

- Application
 - Training API
 - Labeling Section
- Training
 - Transformer-based Tabular (e.g. TabNet)
 - Extend to images, sequences
- Stability & Scalability
 - Monitoring
 - Additional testing
 - Dockerization
 - Better data versioning (currently hard-copies)
 - DAG Orchestration

Thank You
for a great course =)