Harvard

Yá'át'ééh 👋

EASI-22

Edge AI Summer Institute 2022

with Navajo Tech

IEEE COMPUTER SOCIETY

Hi! I'm Brian!

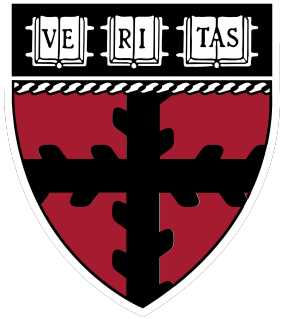
I'm an Assistant Professor of Computer Science
at **Barnard College, Columbia University**

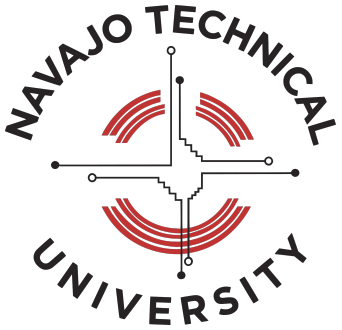
Our team!






with help from **many more**





Our website!

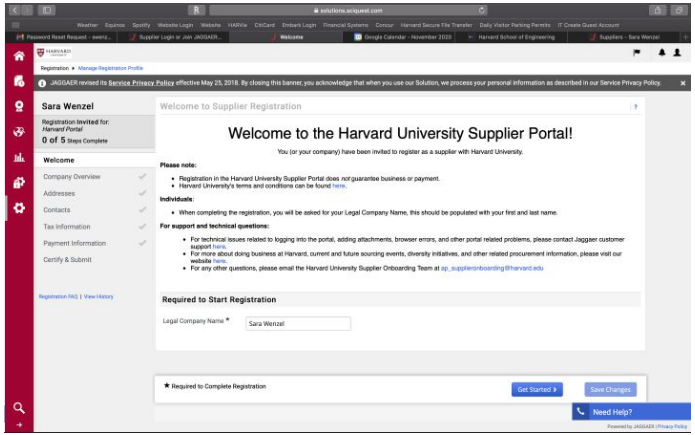
tinyMLedu.org/EASI-22

home base for **all information**!

Make Sure to Pick Up an Arduino Kit!

Question? Contact:
Monsuru Ramoni
mramoni@navajotech.edu

Teachers Sign up for Buy2Pay



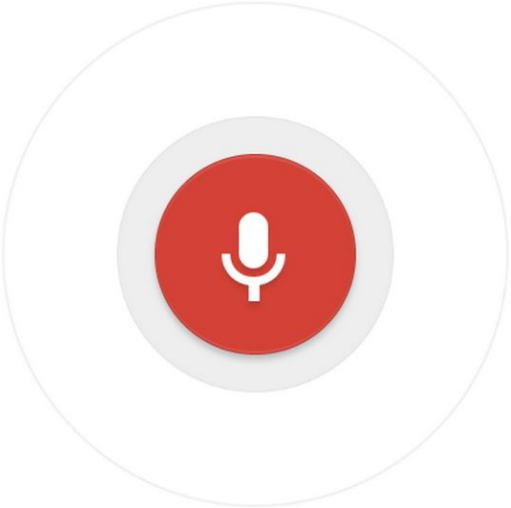
Question? Contact:
Molly Marshall
mmarshall@seas.harvard.edu

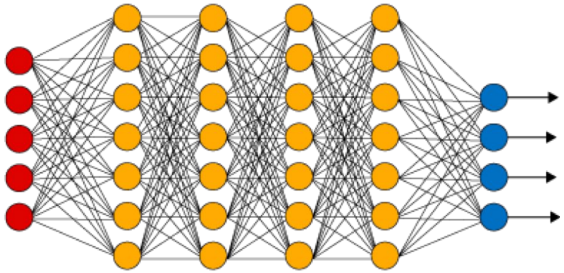
Workshop **Agenda**

Day 1	Introduction to AI and (Tiny)ML	Cloud ML
Day 2	Keyword Spotting for the Navajo Language	Mobile ML
Day 3	Bringing AI/ML from the Cloud to the Edge	Embedded ML

Keyword Spotting in One Slide

If we **pick a simple task** to only identifying a **few key words** we can then use a **small model** and train it with **little data** and fit it onto an **embedded device**





By the end of today: **Hands-on Keyword Spotting (KWS)**

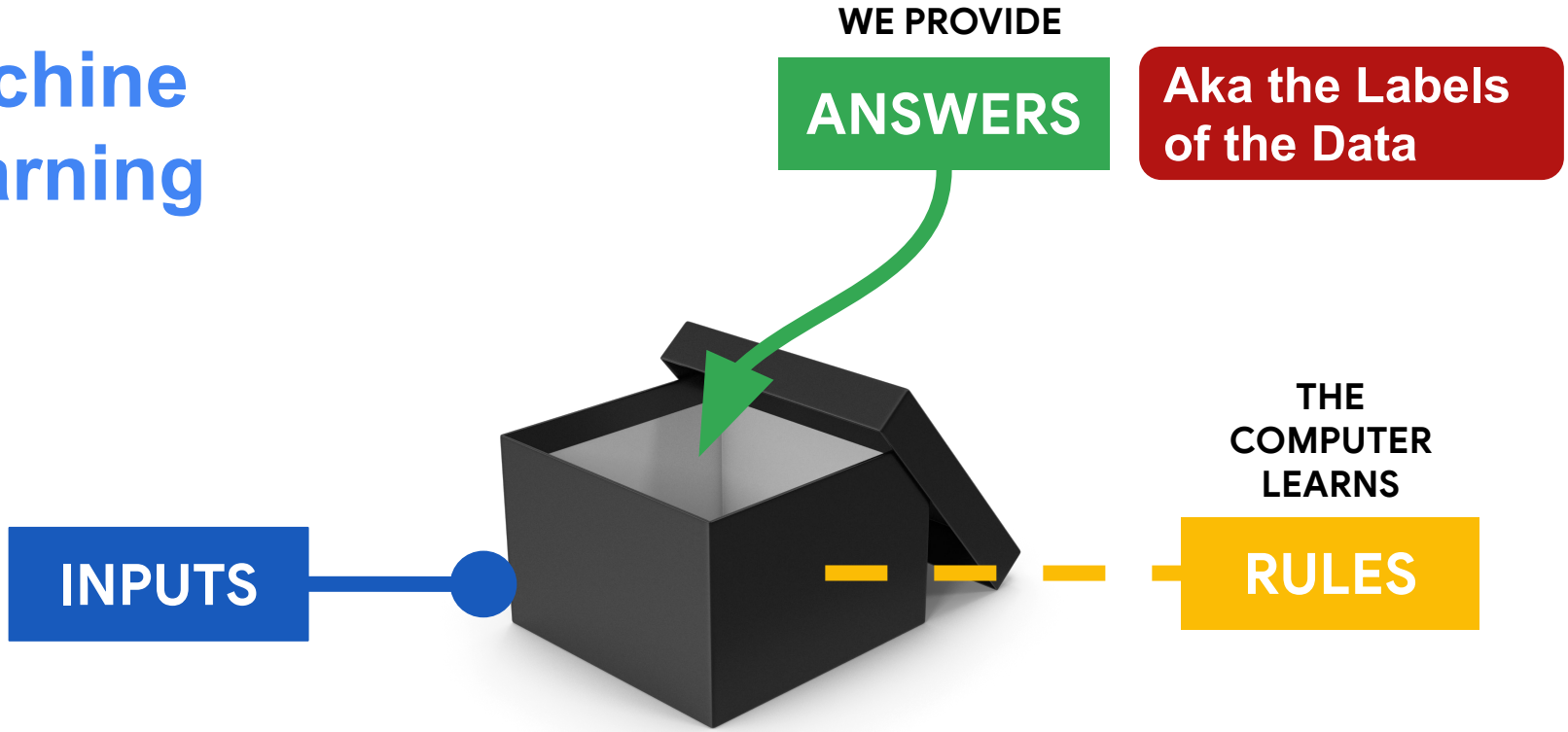
We will explore the **science** behind KWS and **collect data** and **train** our own custom model to recognize “yes” vs. “no” using **Edge Impulse**

Today's Agenda

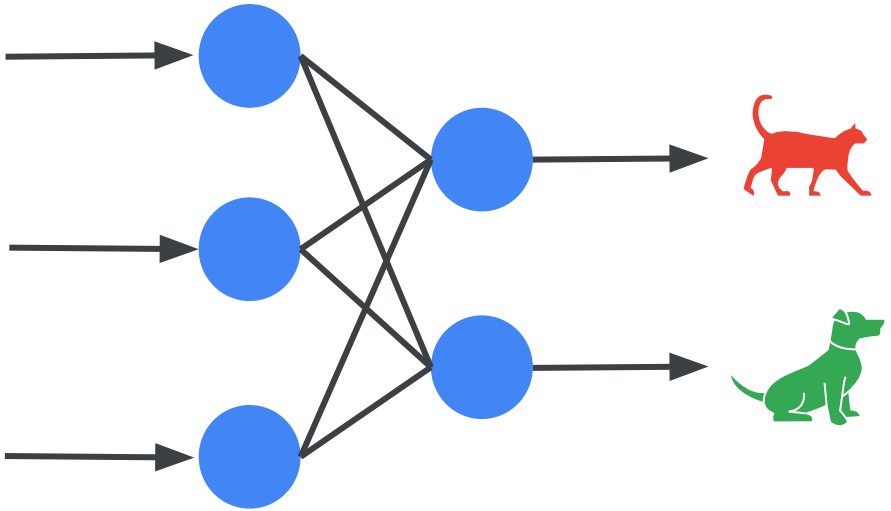
- Preprocessing for Keyword Spotting
- Convolutional Neural Networks for Image Classification
- Hands-on: KWS Data Collection with Edge Impulse
- Hands-on: Training our Model with Edge Impulse
- Hands-on: Testing our Model in the Real World
- Summary

But before we dive into all of that – a little quick review!

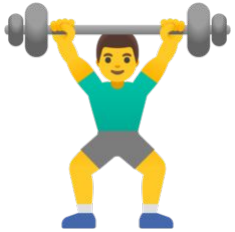
Machine Learning

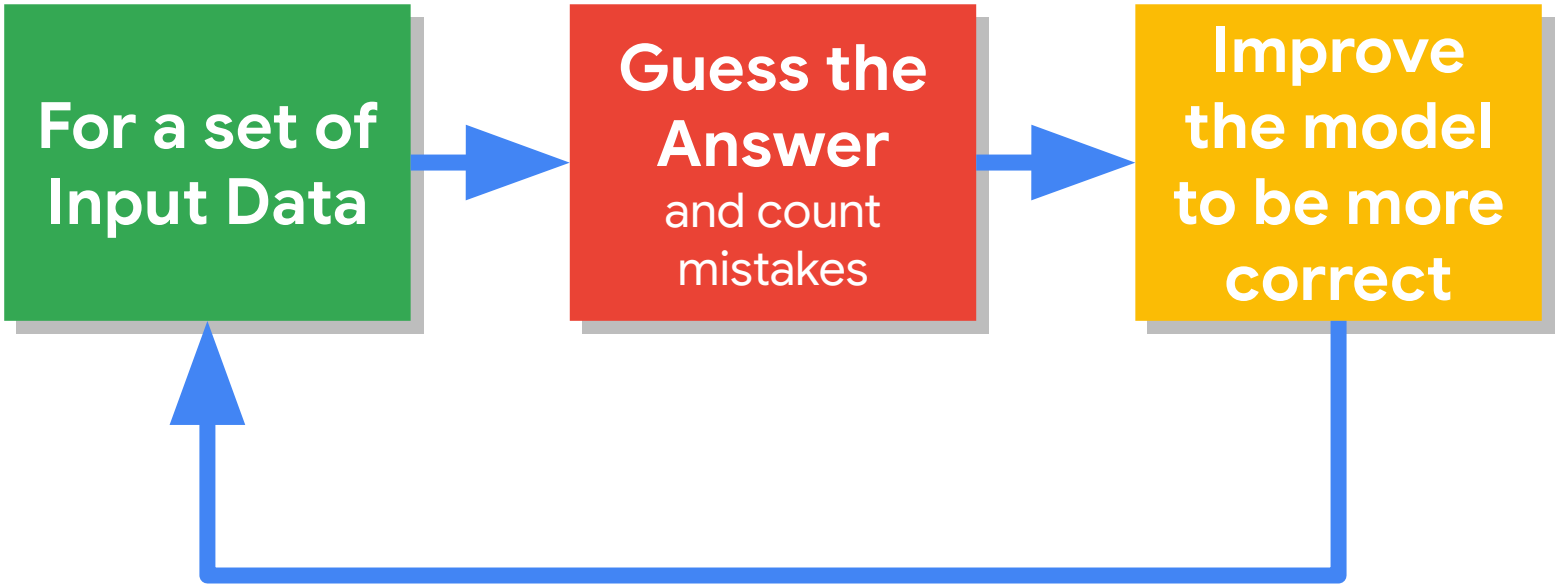


Machine Learning with **neural networks**?

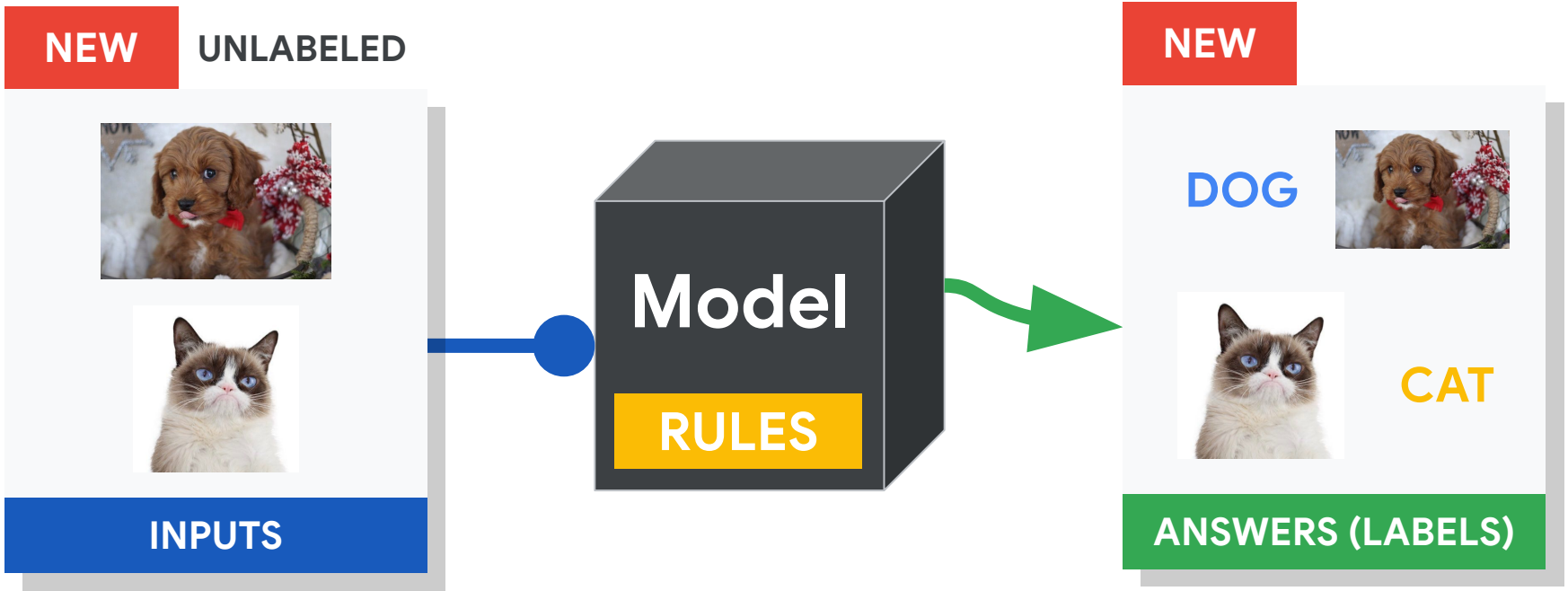


Training the machine

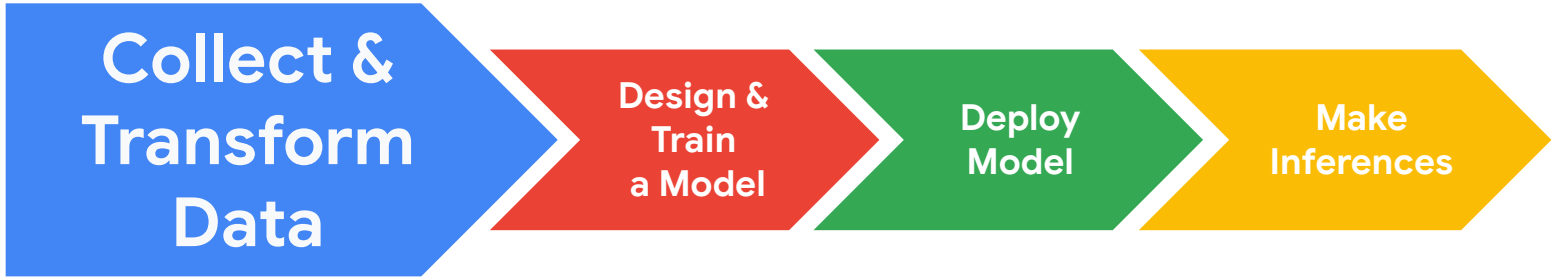




After it's **learned** use it for **inference**:



The (Tiny) Machine Learning **Workflow**

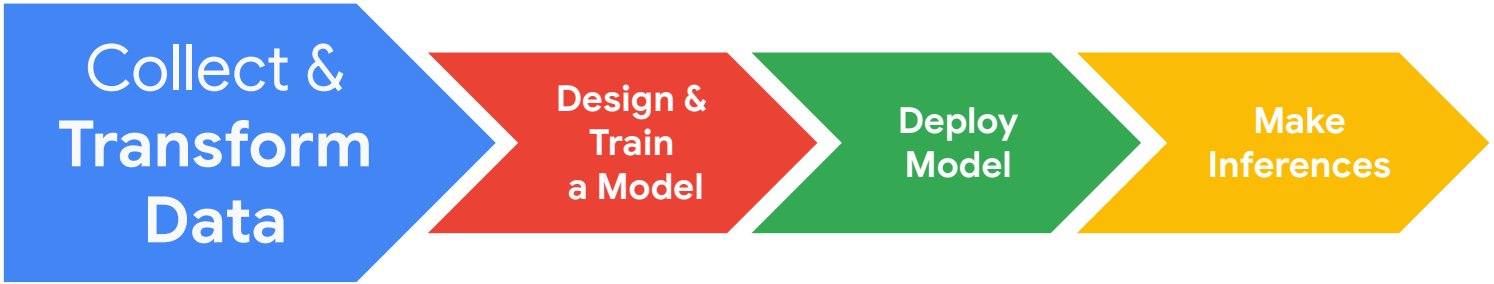


If ML is going to be everywhere we need to consider how to best collect GOOD data RESPONSIBLY

Today's Agenda

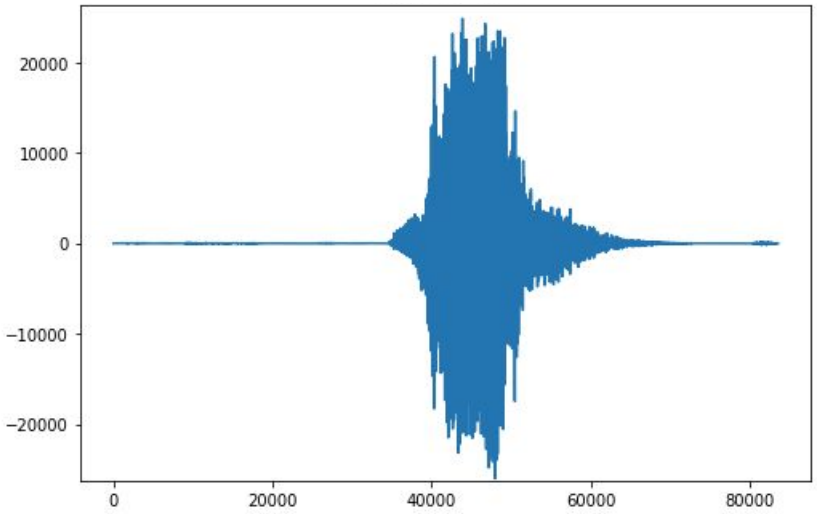
- **Preprocessing for Keyword Spotting**
- Convolutional Neural Networks for Image Classification
- Hands-on: KWS Data Collection with Edge Impulse
- Hands-on: Training our Model with Edge Impulse
- Hands-on: Testing our Model in the Real World
- Summary

Machine Learning **Workflow**



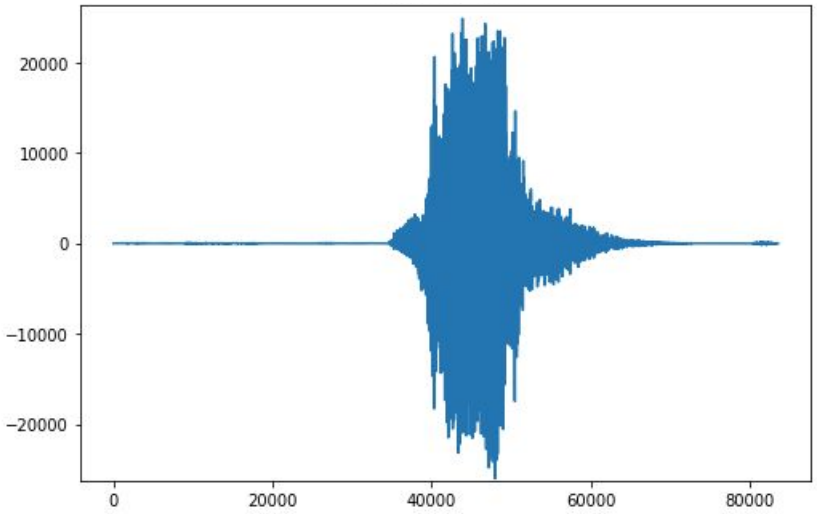
This is an audio signal

“Yes” *(spoken loudly)*

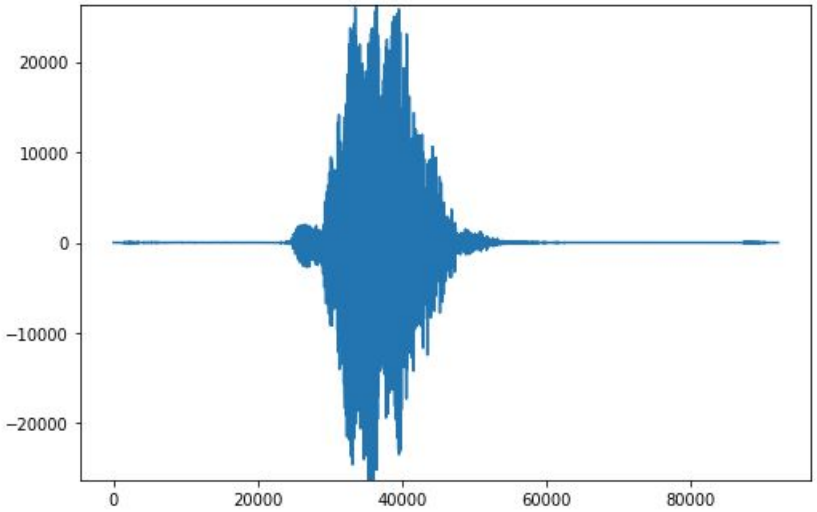


Can you tell these two signals apart?

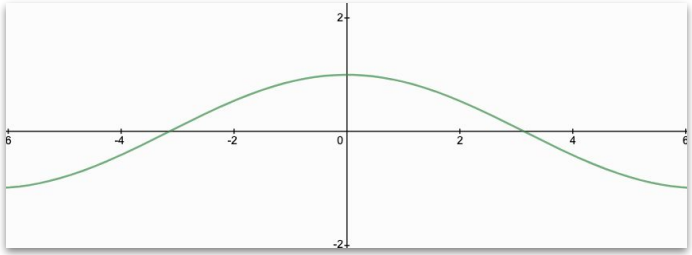
“Yes” *(spoken loudly)*



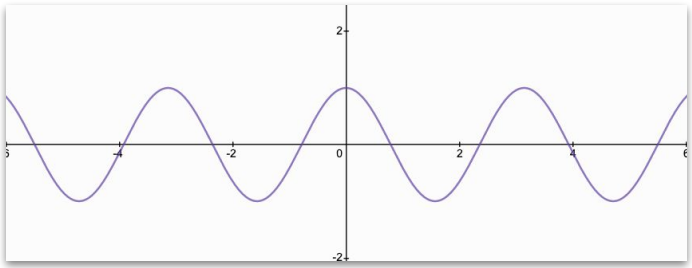
“No” *(spoken loudly)*



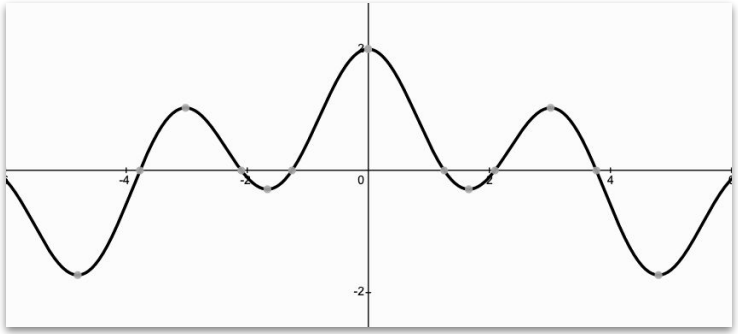
Signal **Components**?



+



=



Signal **Components**?

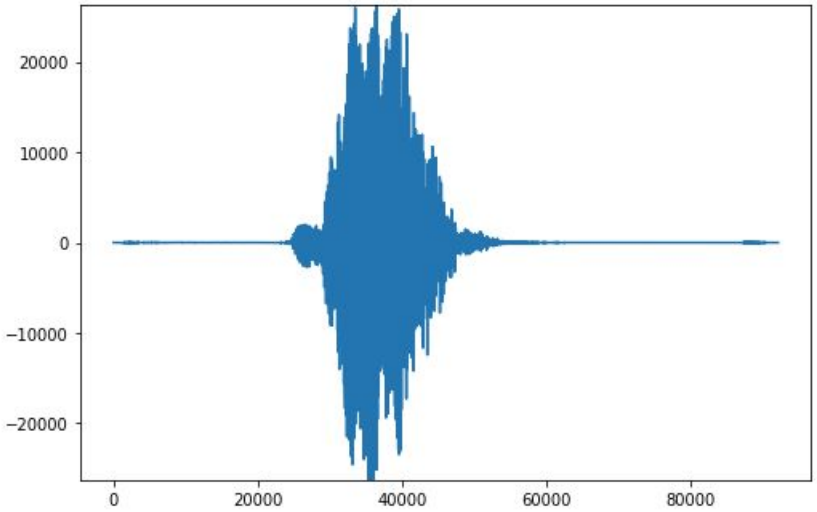
?

+

?

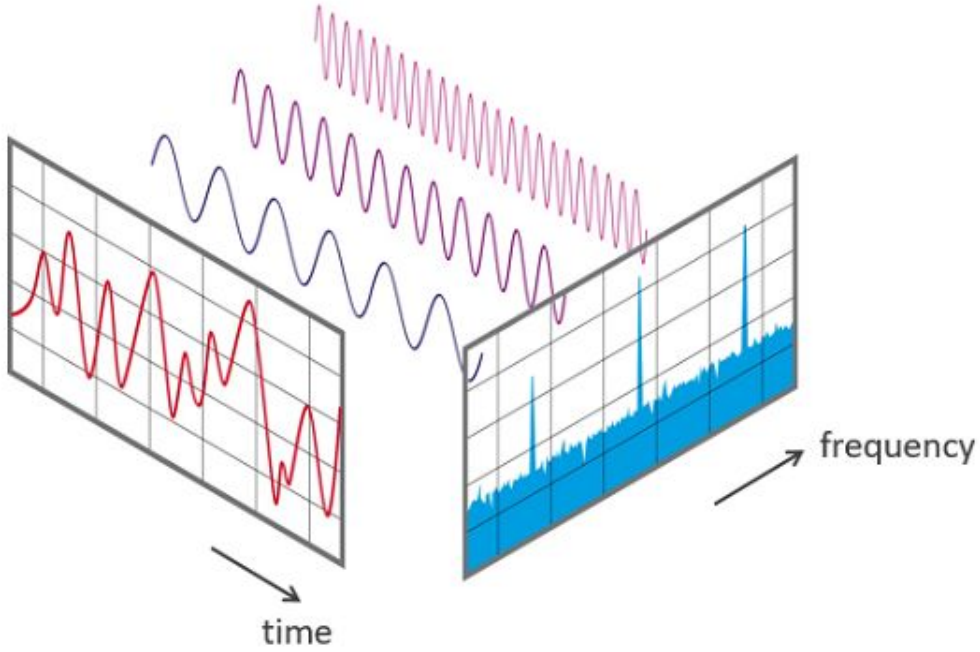
=

“No” *(spoken loudly)*

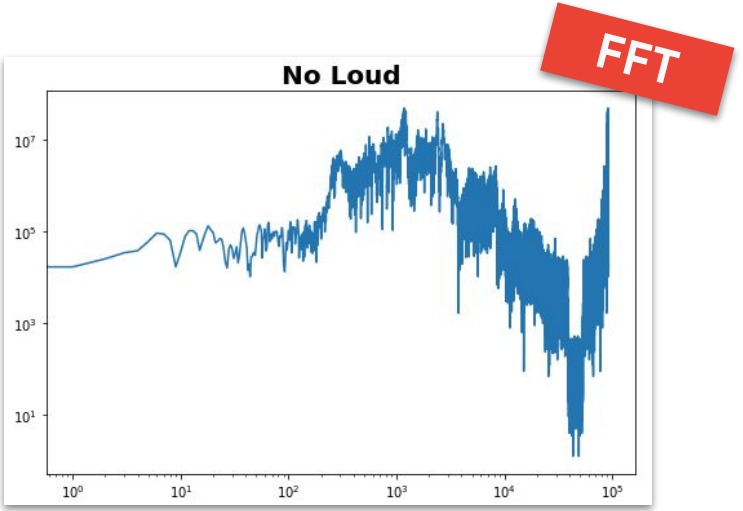


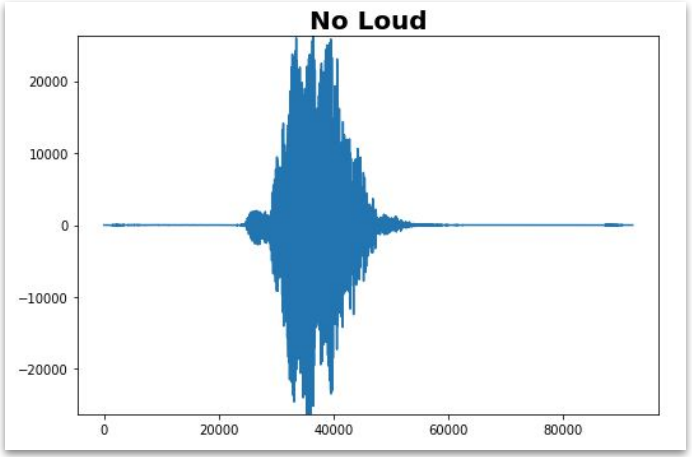
Fast Fourier Transform:

extract the frequencies from a signal



Fast Fourier Transform

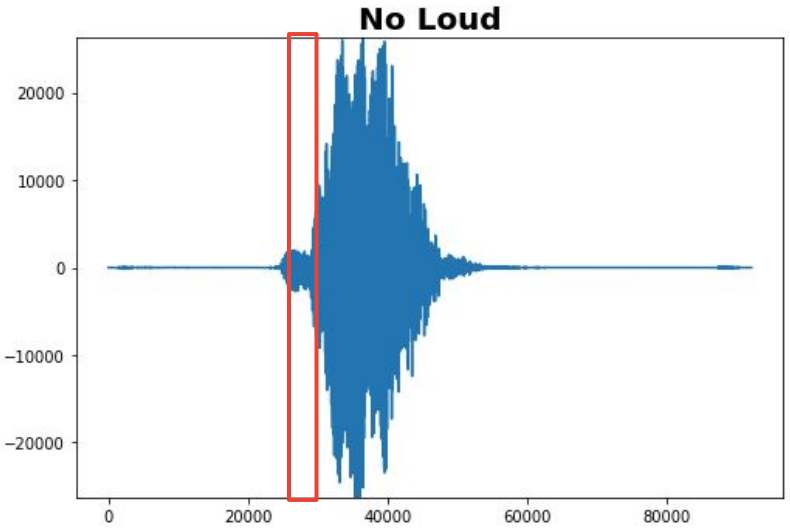




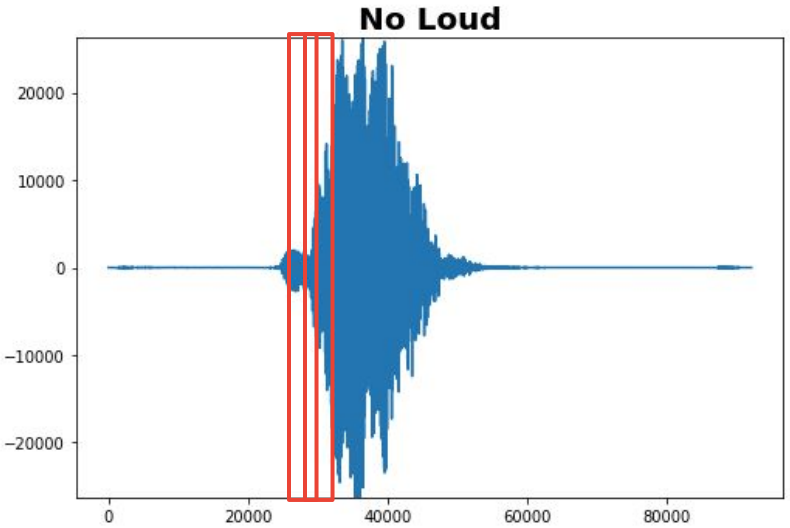
Frequency

Time

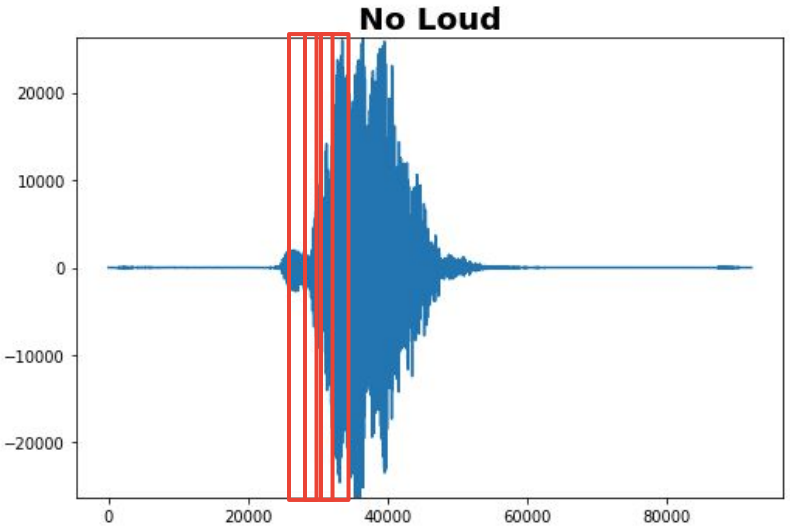
Building a **Spectrogram** using FFTs



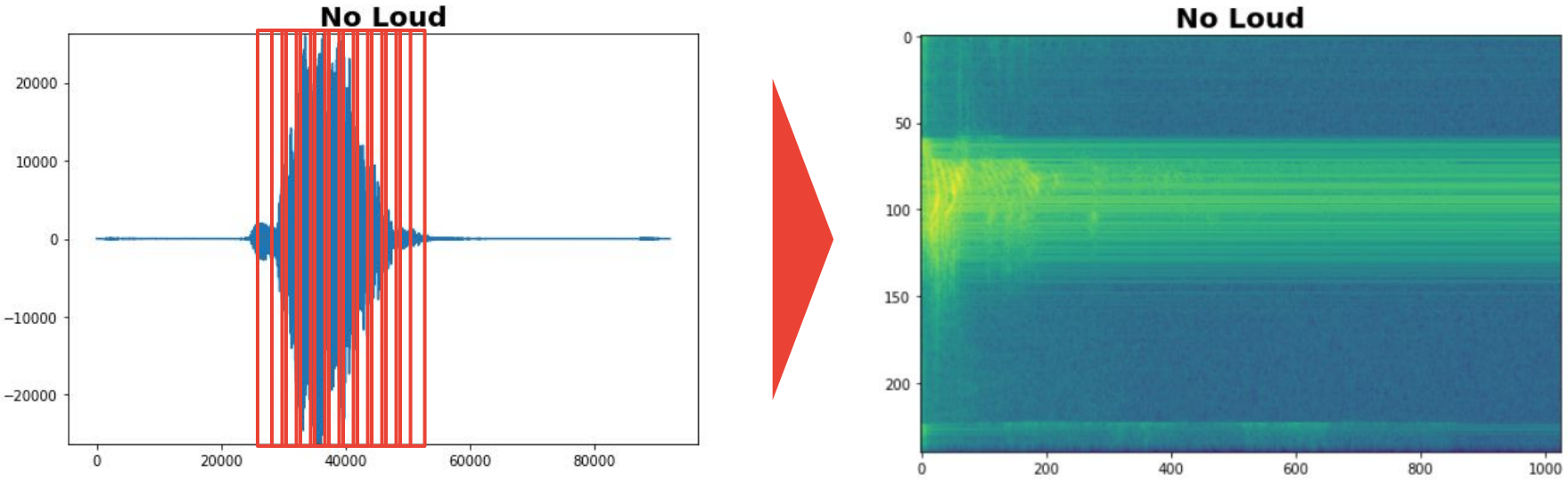
Building a **Spectrogram** using FFTs



Building a **Spectrogram** using FFTs

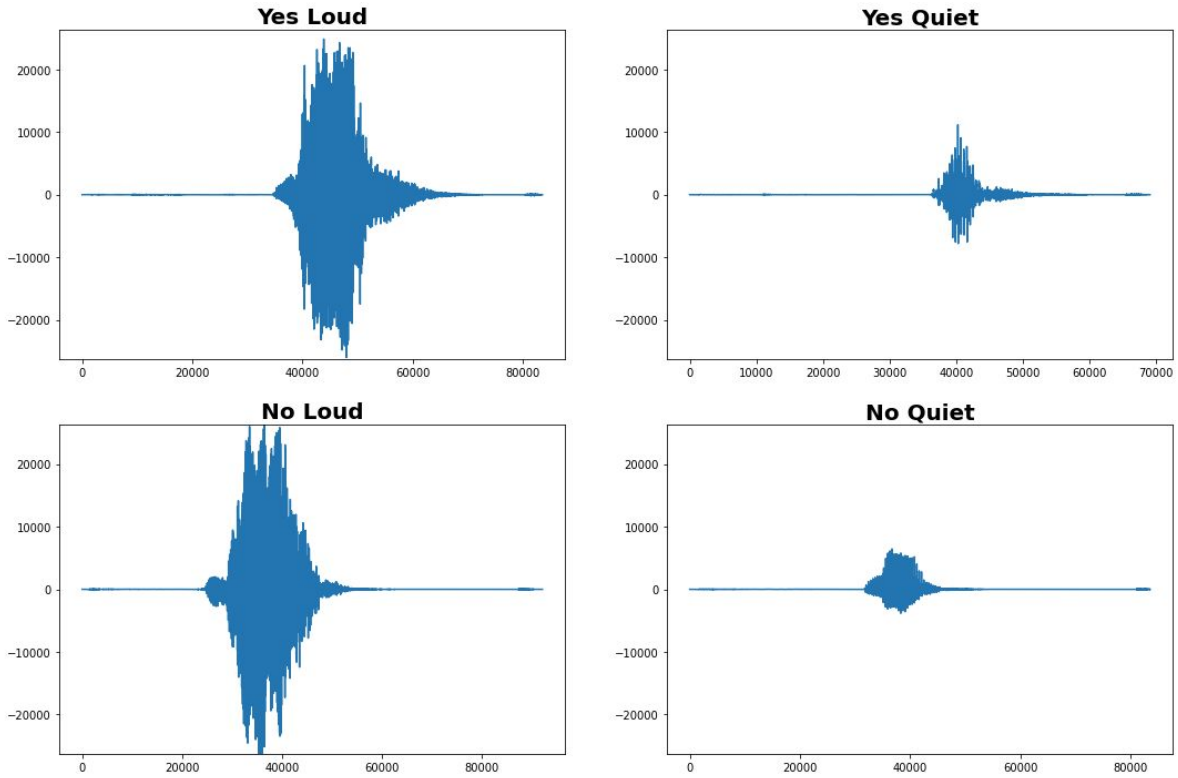


Building a **Spectrogram** using FFTs

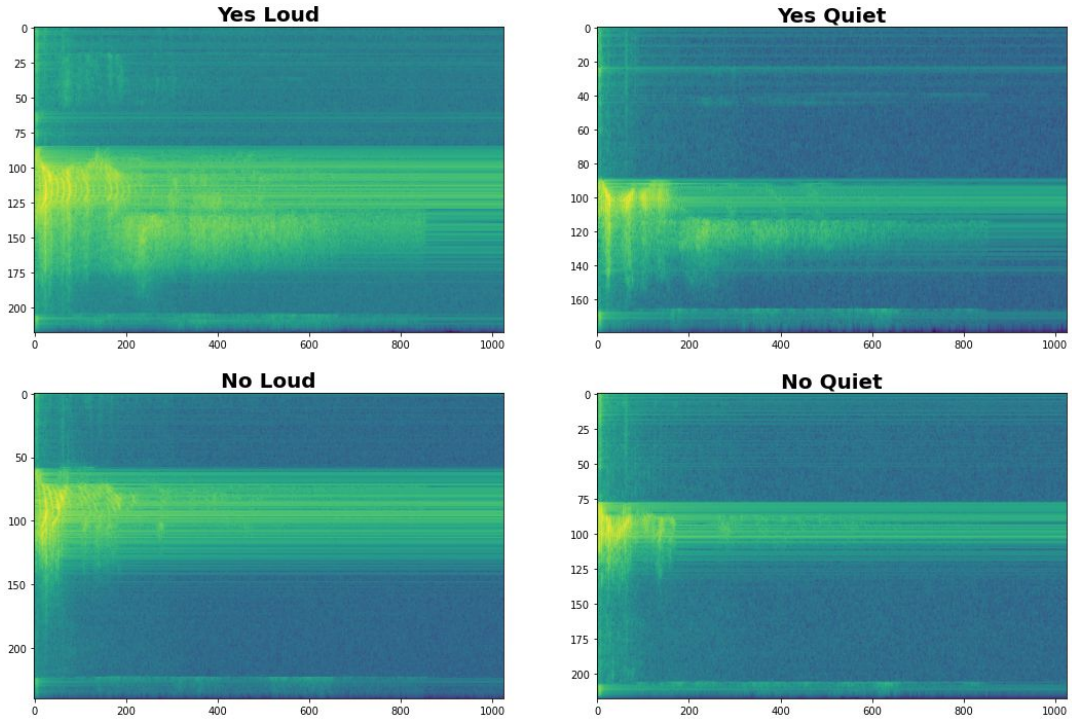


Essentially if you **stack up all the FFTs in a row** then you get the **Spectrogram** (time vs. frequency with color indicating intensity)

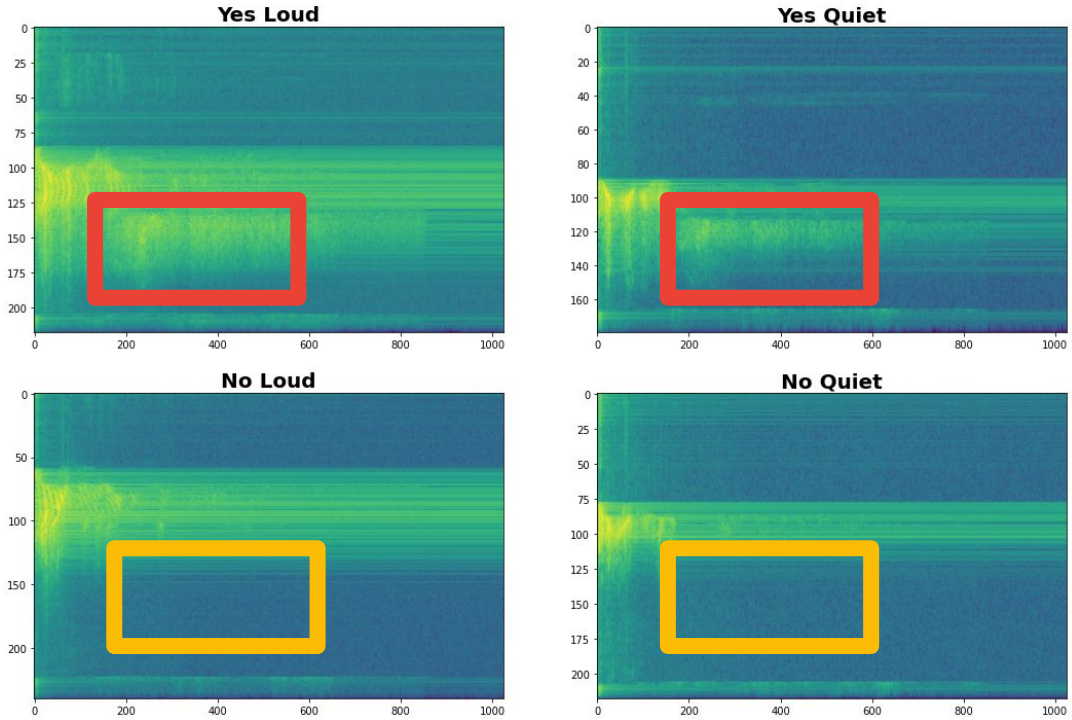
Spectrograms help differentiate the data



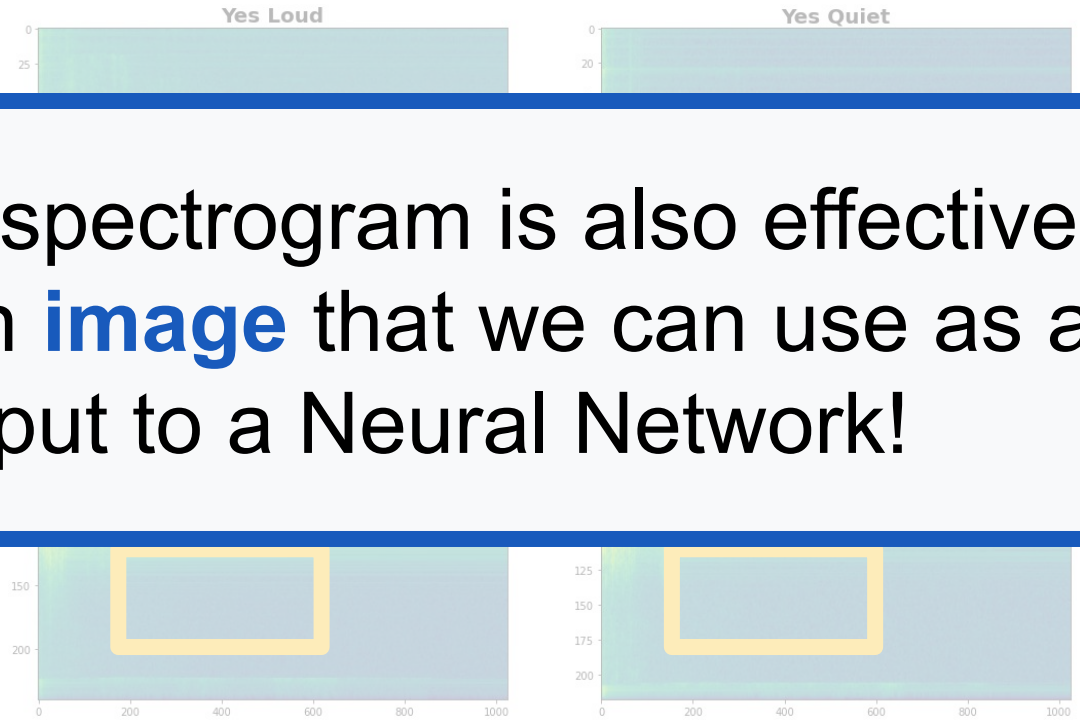
Spectrograms help differentiate the data



Spectrograms help differentiate the data



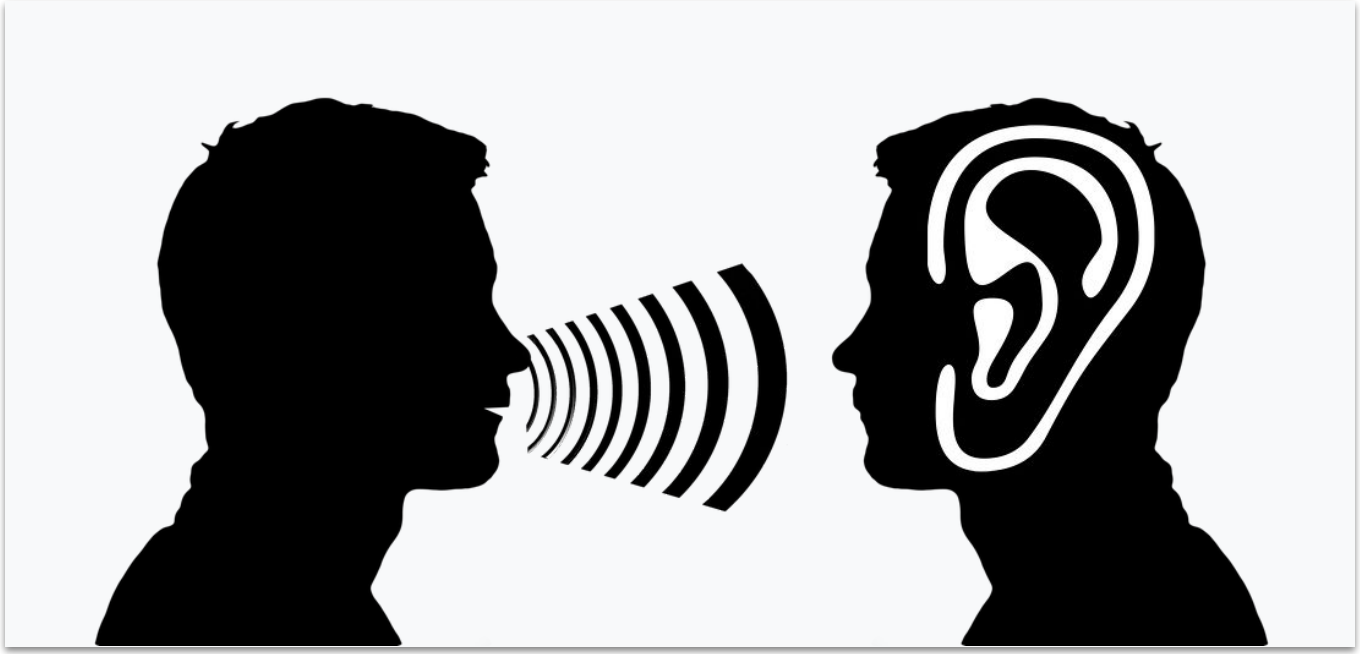
Data Preprocessing: **Spectrograms**



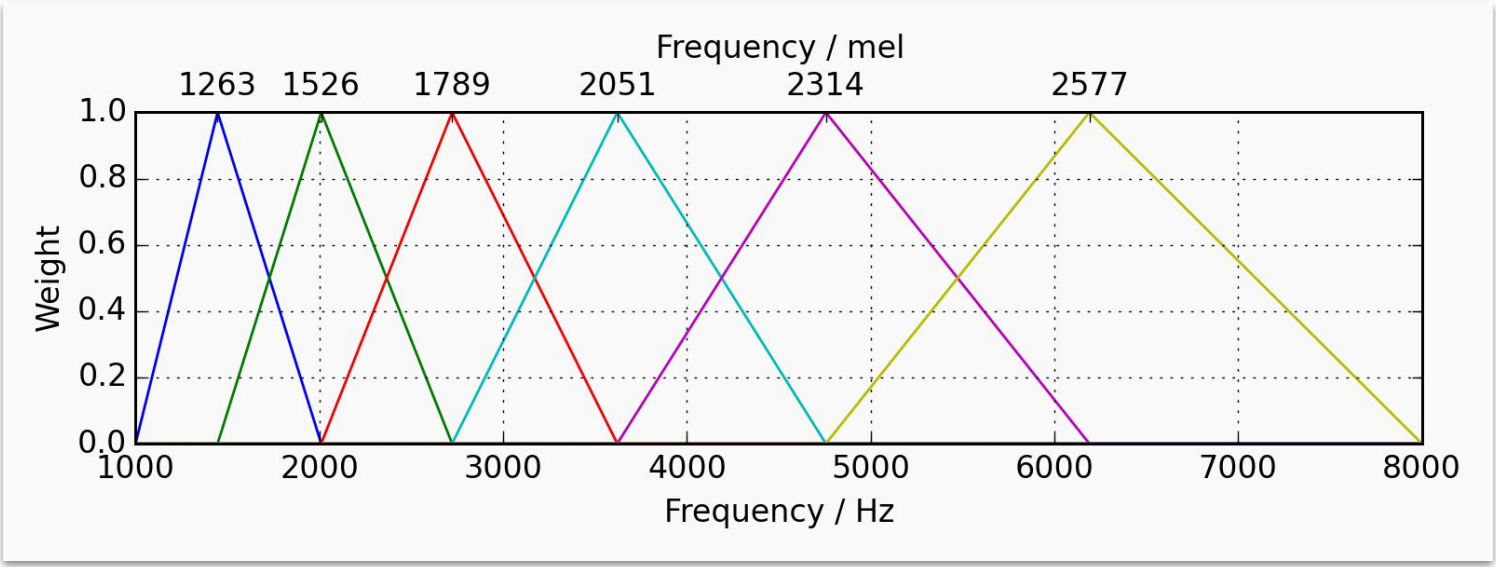
A spectrogram is also effectively an **image** that we can use as an input to a Neural Network!

Can we do **better** than a spectrogram?

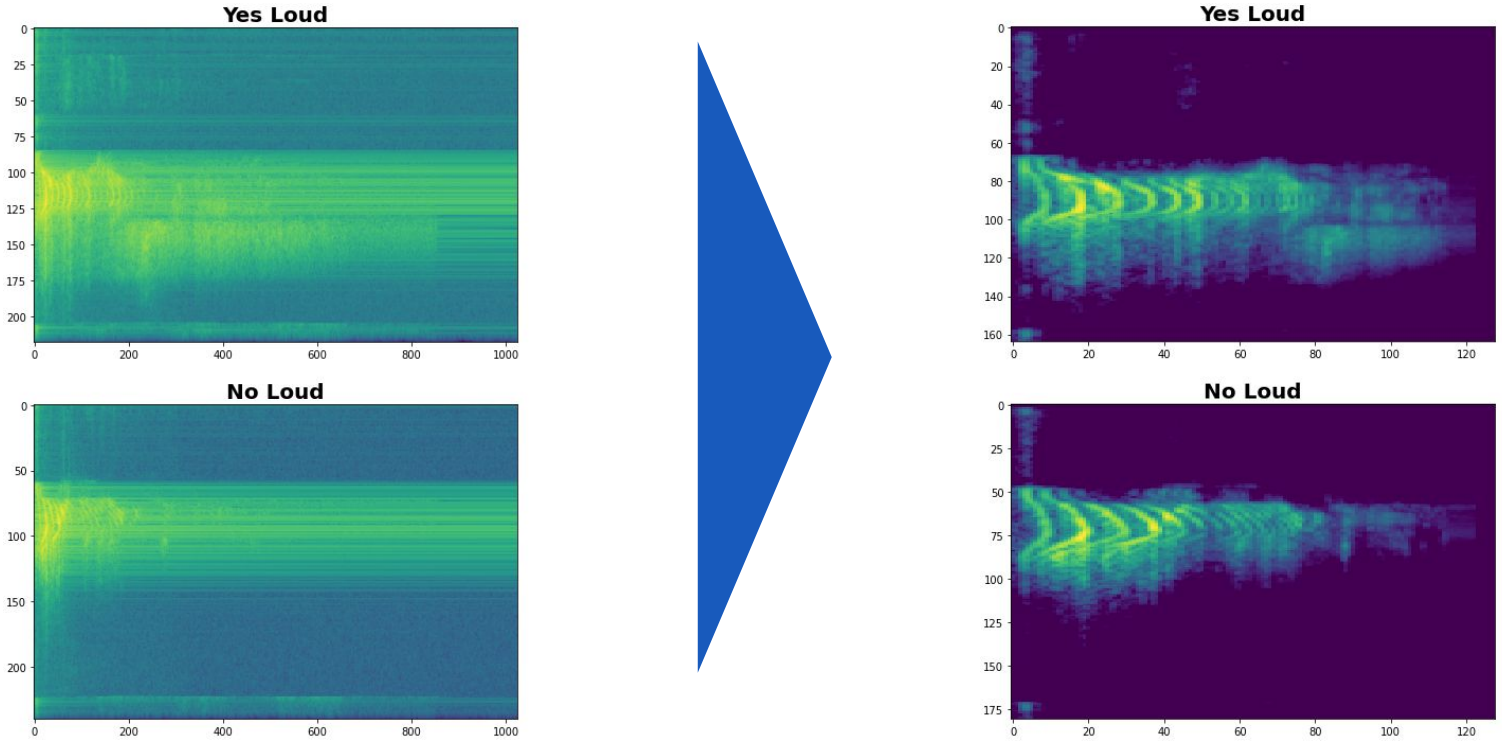
Can we take **domain knowledge** into account?



Mel Filterbanks

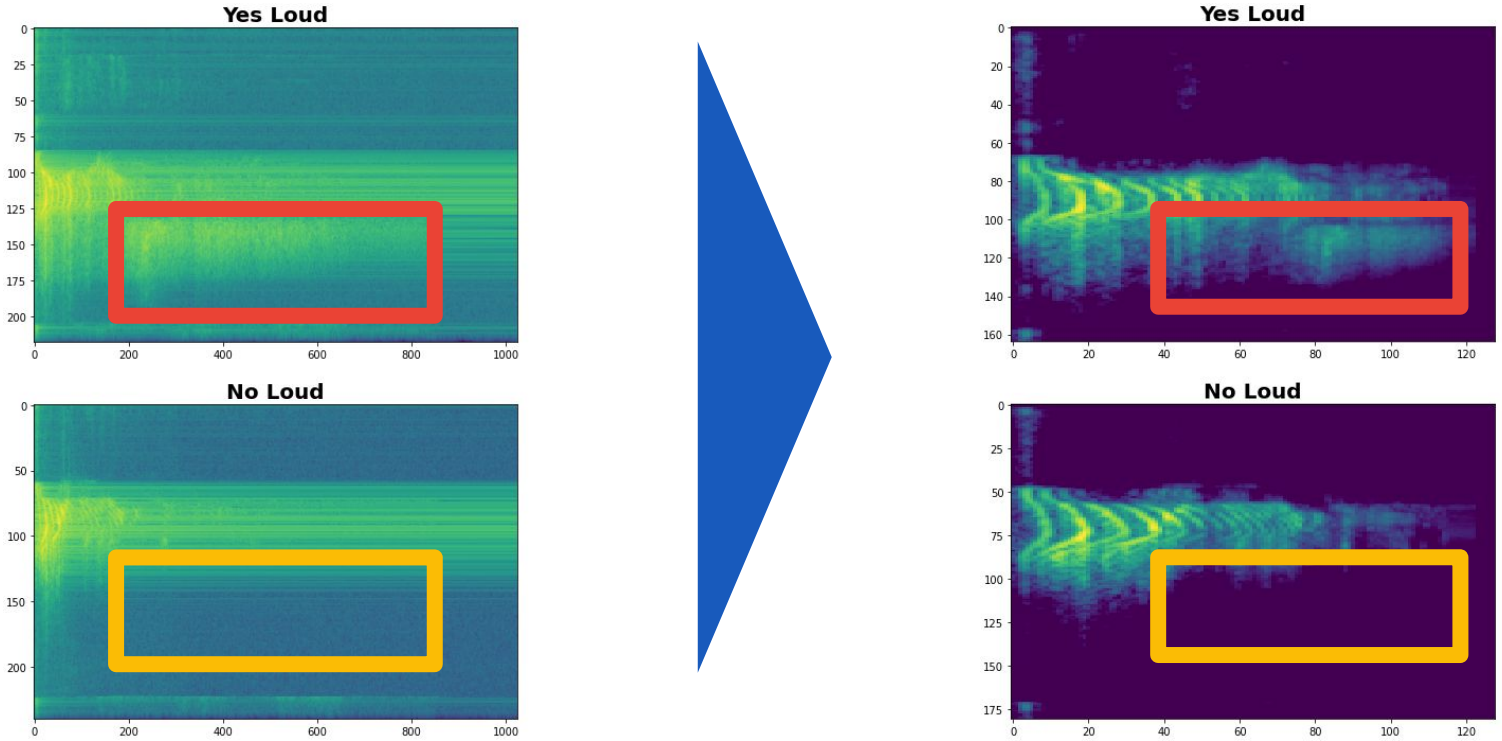


Spectrograms v. MFCCs



Colab Link

Spectrograms v. MFCCs



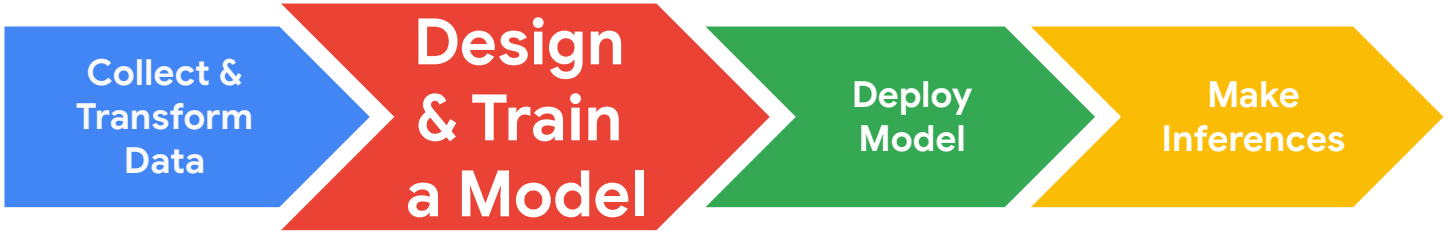
Today's Agenda

- **Preprocessing for Keyword Spotting**
- Convolutional Neural Networks for Image Classification
- Hands-on: KWS Data Collection with Edge Impulse
- Hands-on: Training our Model with Edge Impulse
- Hands-on: Testing our Model in the Real World
- Summary

Today's Agenda

- Preprocessing for Keyword Spotting
- **Convolutional Neural Networks for Image Classification**
- Hands-on: KWS Data Collection with Edge Impulse
- Hands-on: Training our Model with Edge Impulse
- Hands-on: Testing our Model in the Real World
- Summary

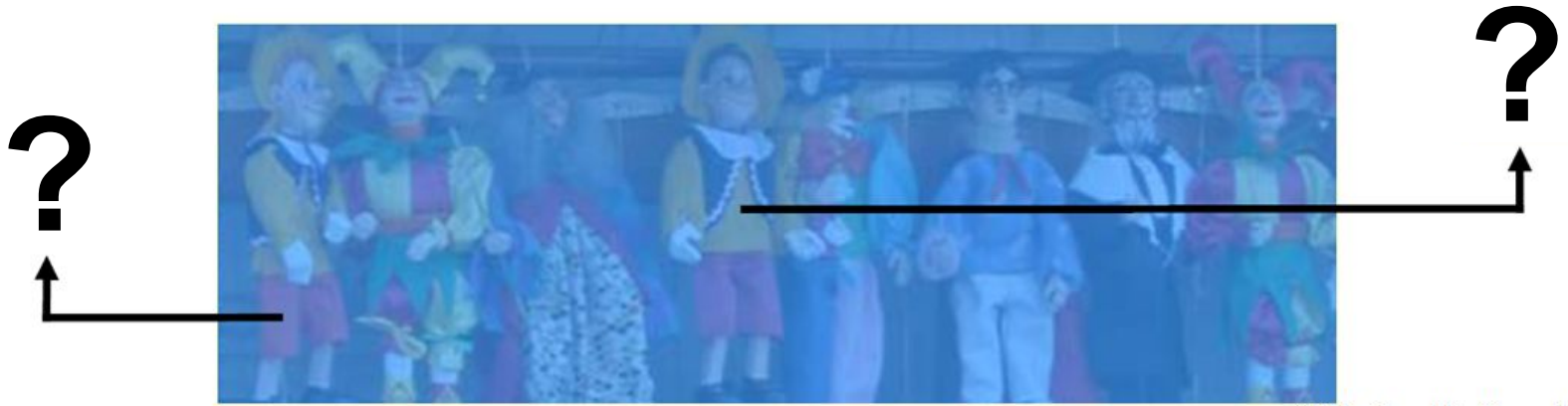
Machine Learning **Workflow**



Computer Vision is Hard

Computer Vision is Hard

What color are the pants and the shirt?



Slide Credit: Hamilton Chong

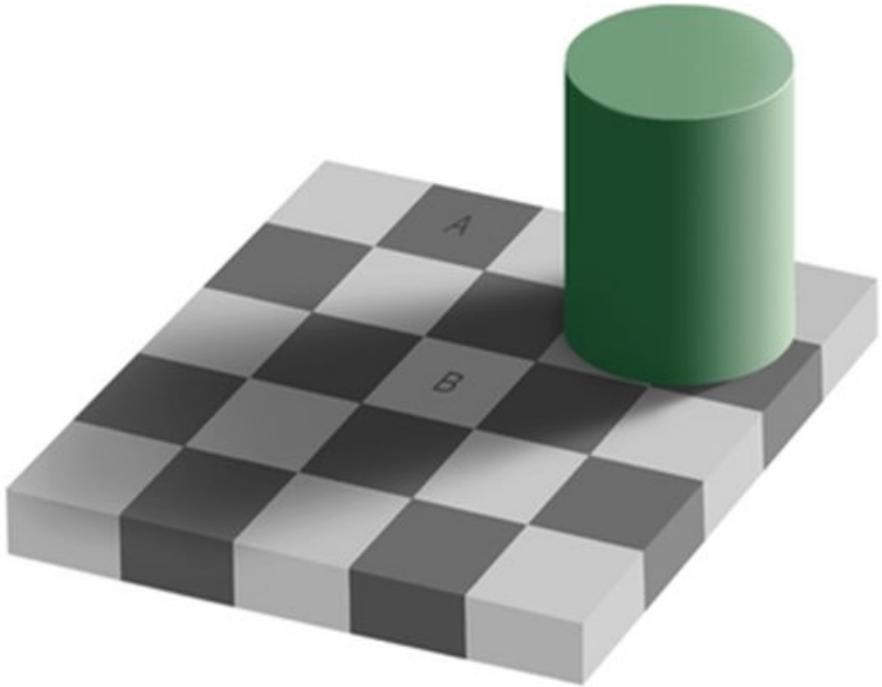
Computer Vision is Hard

Slide Credit: Hamilton Chong

Computer Vision is Hard

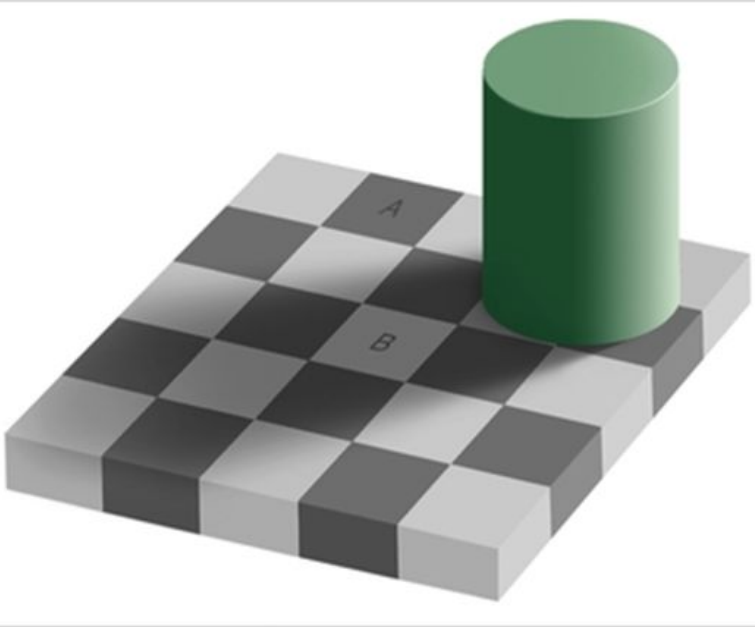
Slide Credit: Hamilton Chong

Computer Vision is Hard

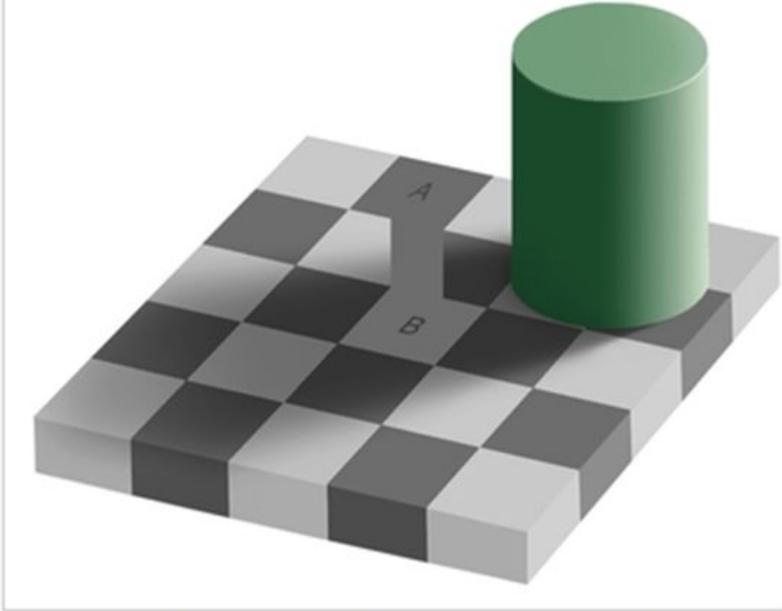


Is square A or B darker in color?

Computer Vision is Hard

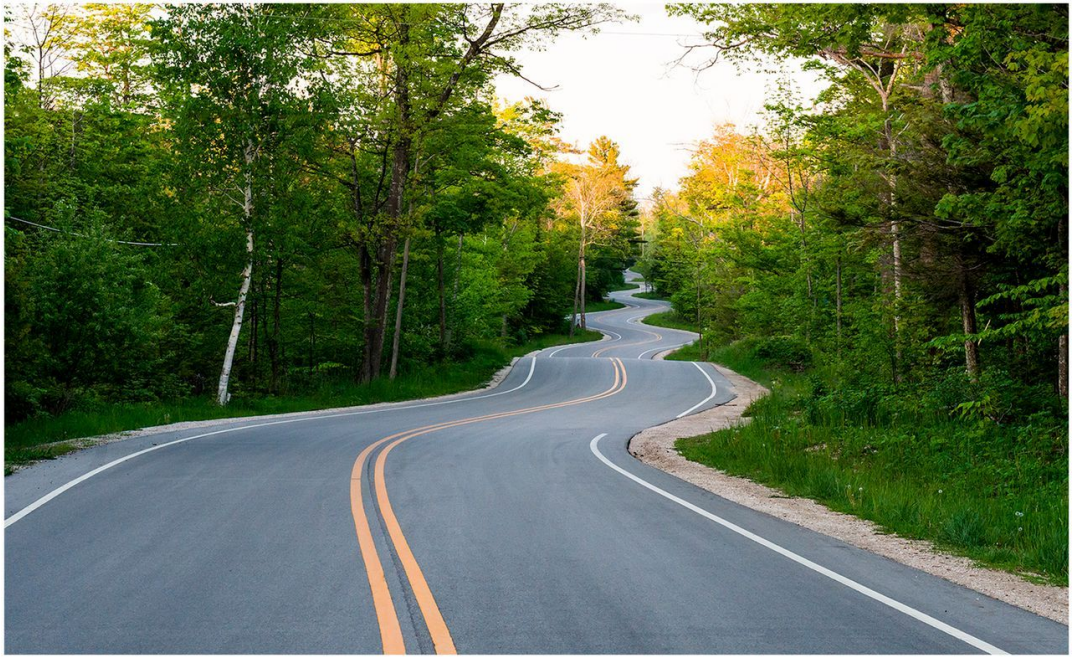


Areas of the image A and B are the same color



A rectangle of the same color has been drawn connecting the two areas of the image

What **Features** of the image might be important for self driving cars?

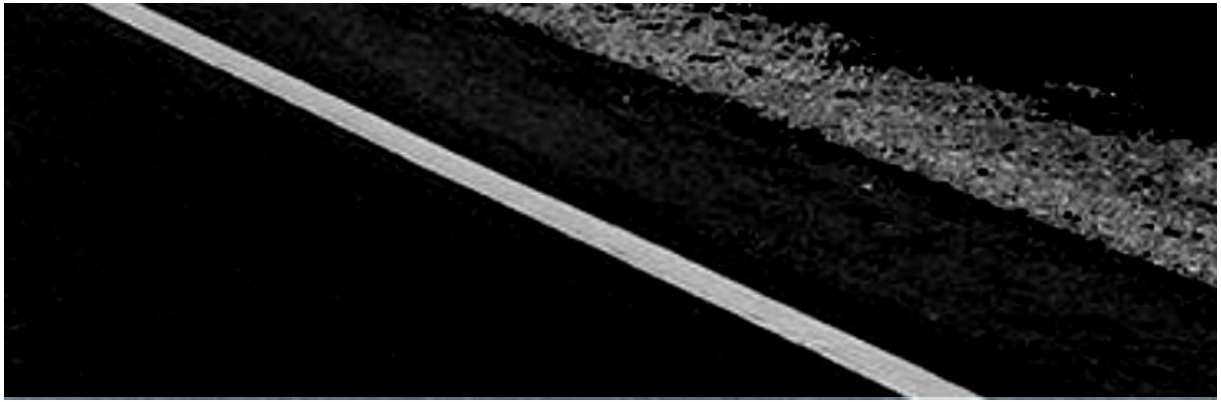


What **Features** of the image might be important for self driving cars?

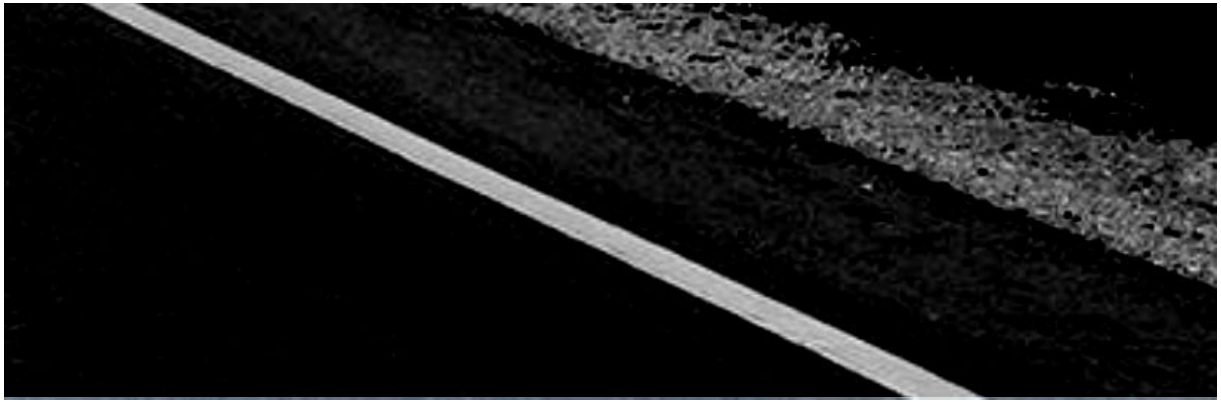
Maybe straight lines to see the lanes of the road?

How might we find these features?

How might we find these features?

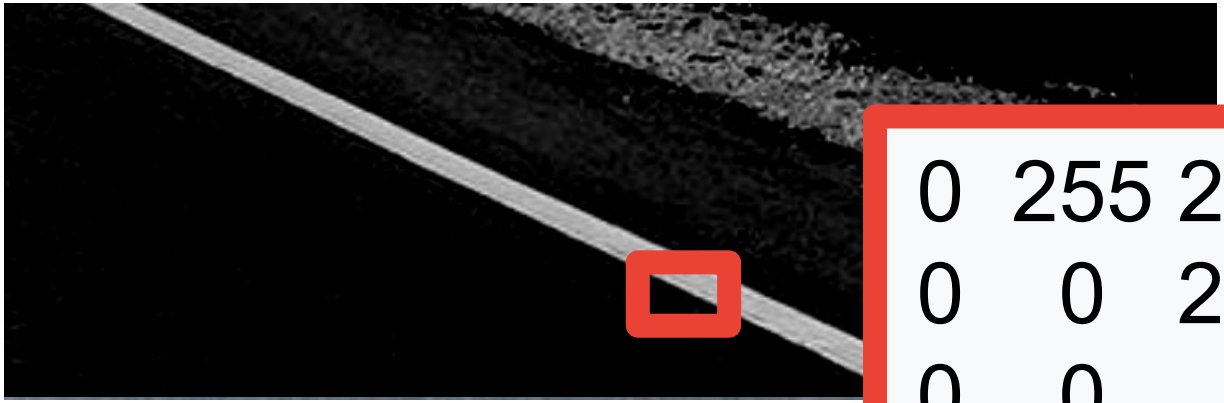


How might we find these features?



Black: 0
White: 255

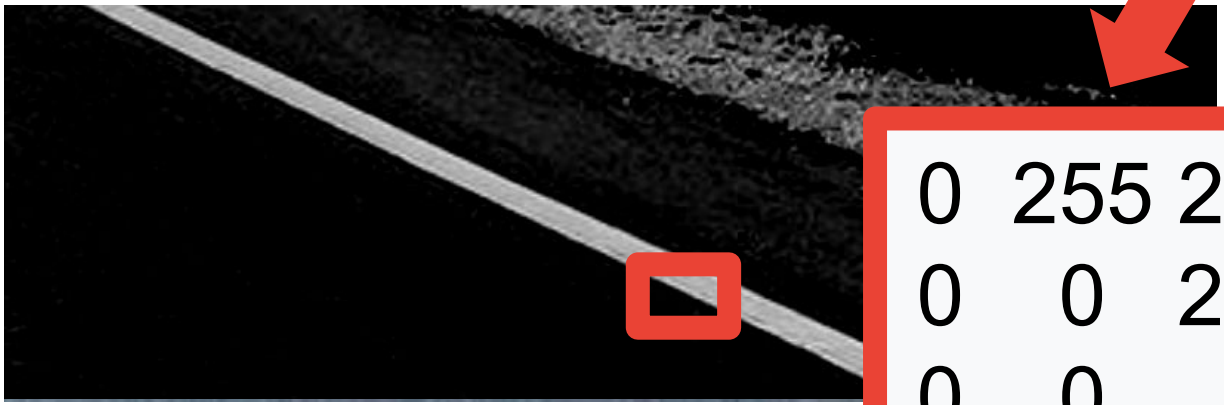
How might we find these features?



0	255	255	255	255
0	0	255	200	255
0	0	0	255	255
0	0	0	0	255
0	0	0	0	0

Black: 0
White: 255

How might we find these features?



Look for a Big Change!

0	255	255	255	255
0	0	255	200	255
0	0	0	255	255
0	0	0	0	255
0	0	0	0	0

Black: 0
White: 255

How might we find these features?

Convolutions

How might we find these features?

Convolutions

Original Image

0	0	0	255	255	255
0	0	0	255	255	255
0	0	0	255	255	255
0	0	0	255	255	255
0	0	0	255	255	255
0	0	0	255	255	255

How might we find these features?

Convolutions

Original Image

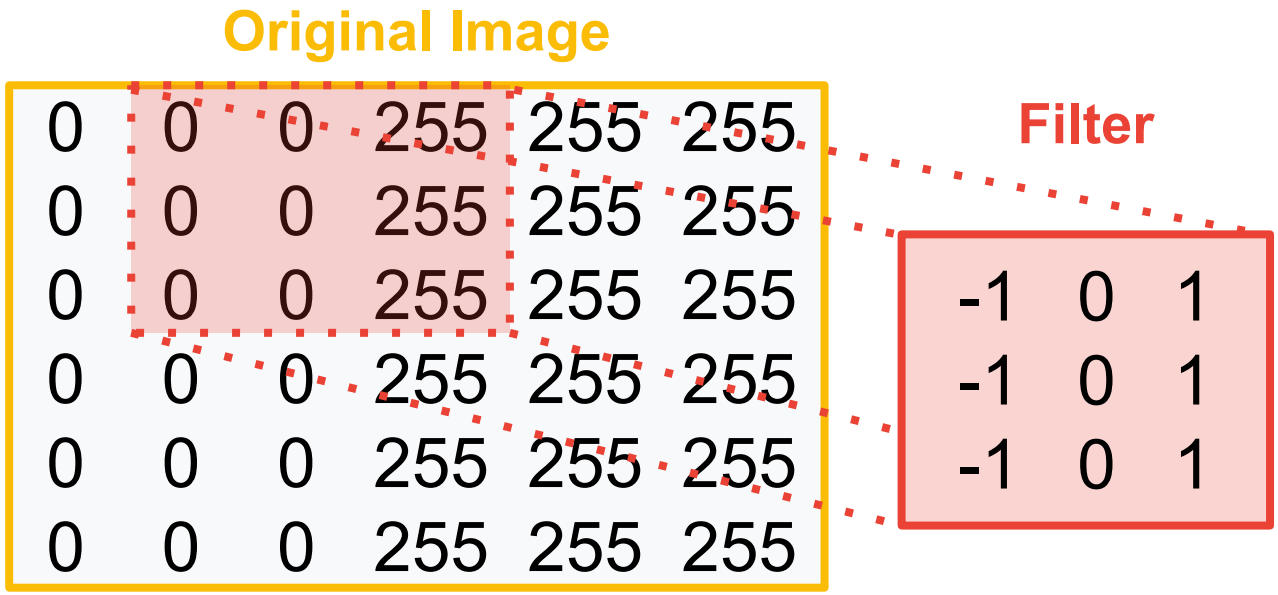
0	0	0	255	255	255
0	0	0	255	255	255
0	0	0	255	255	255
0	0	0	255	255	255
0	0	0	255	255	255
0	0	0	255	255	255

Filter

-1	0	1
-1	0	1
-1	0	1

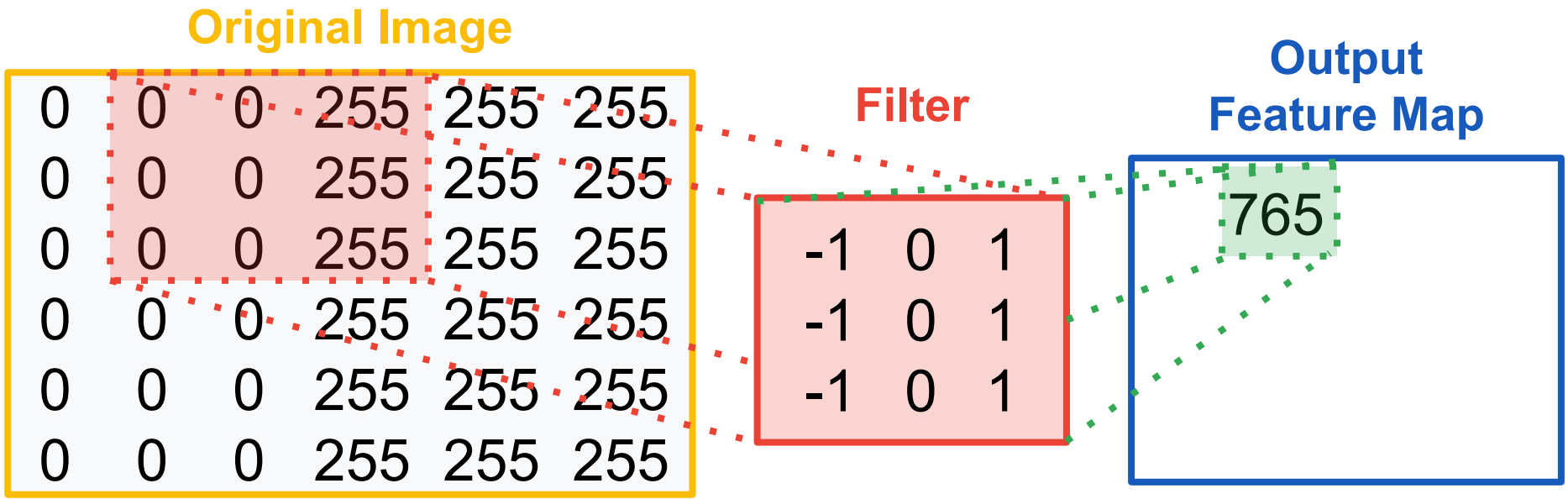
How might we find these features?

Convolutions



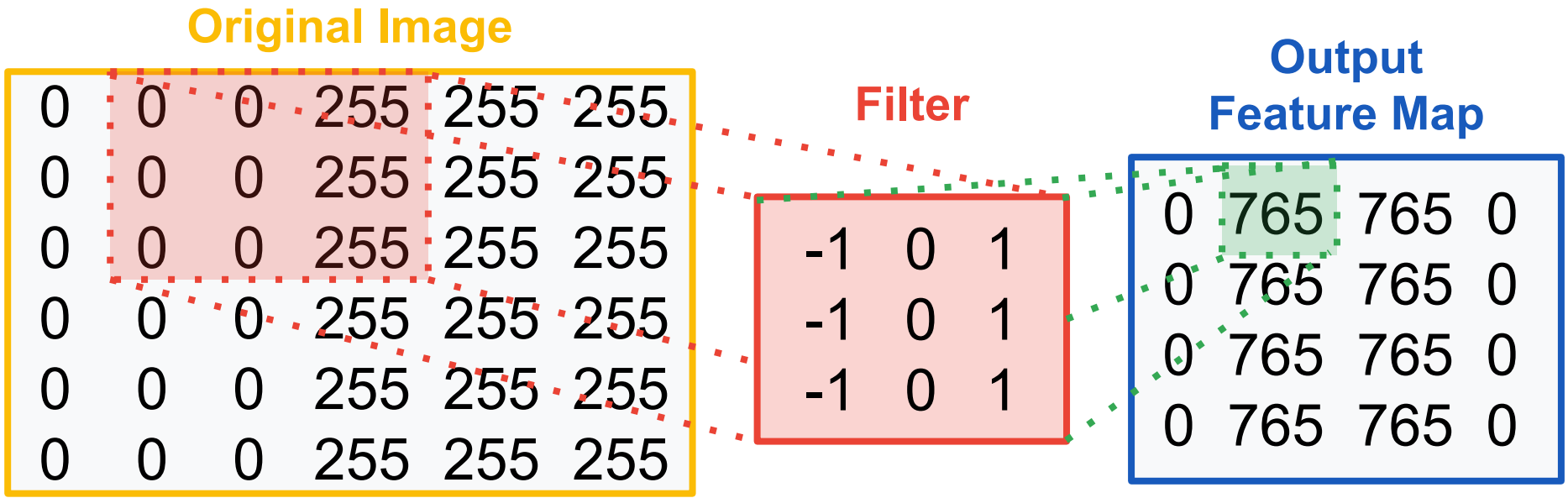
How might we find these features?

Convolutions



How might we find these features?

Convolutions



Colab Link

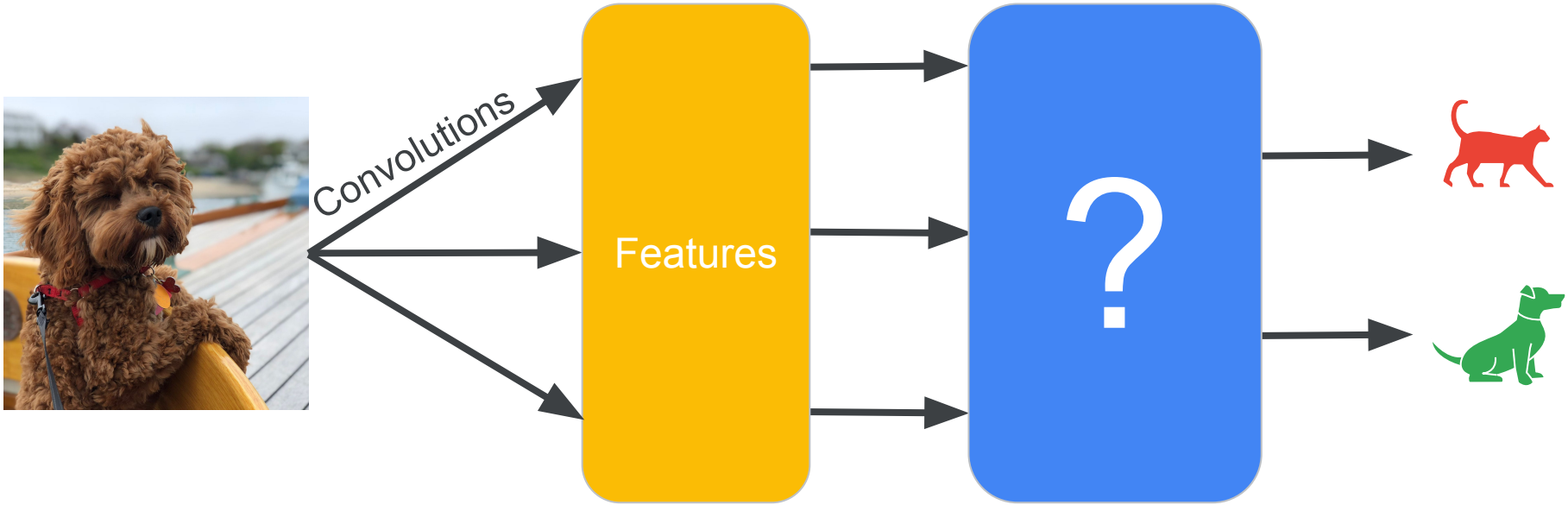
How might we find these features?

Convolutions

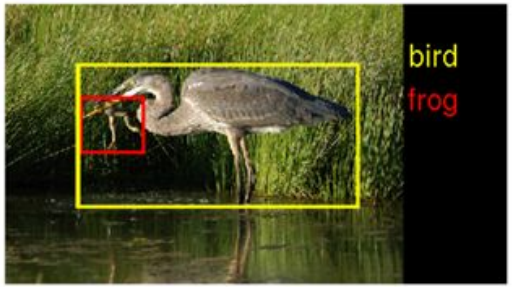
-1	0	1
-2	0	2
-1	0	1

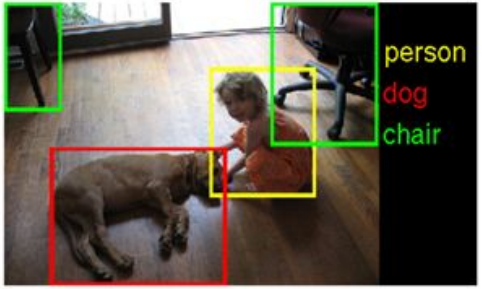
Features

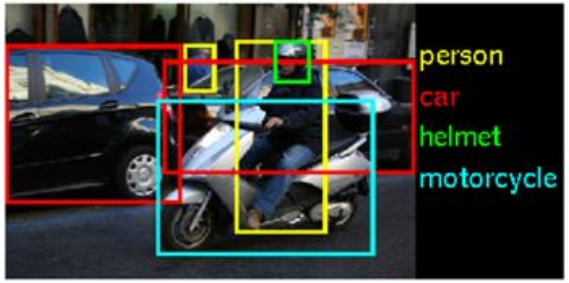
How might we combine these features to **classify an object**?



The ImageNet Challenge and the birth of CNNs



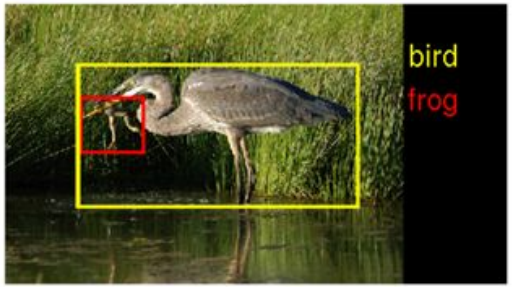


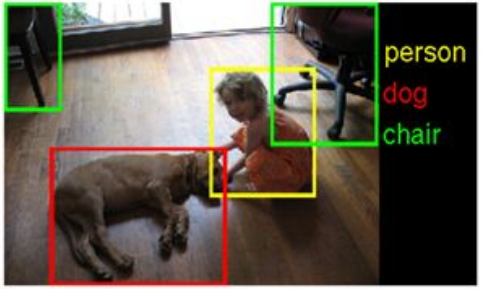


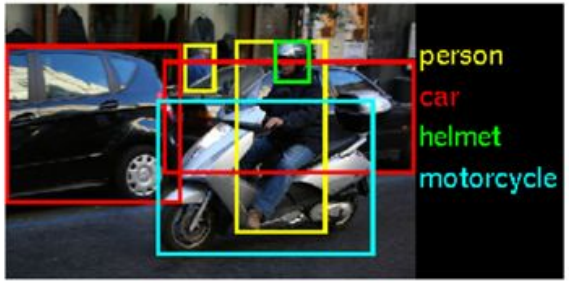
The ImageNet Challenge provided 1.2 million examples of 1,000 **labeled** items and challenged algorithms to learn from the data and then was tested on another 100,000 images

The ImageNet Challenge and the birth of CNNs

The ImageNet Challenge and the birth of CNNs



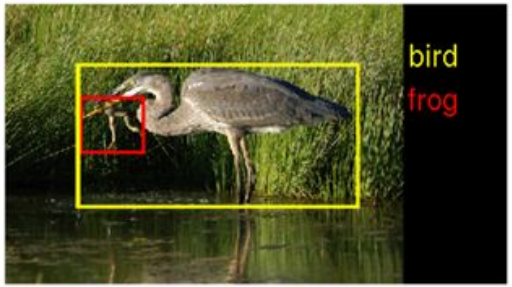


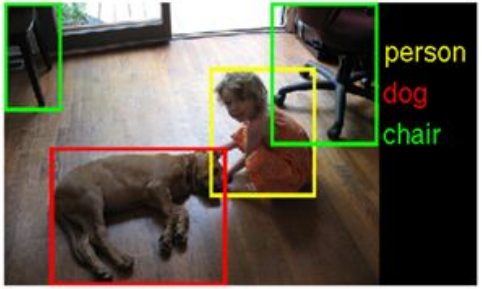


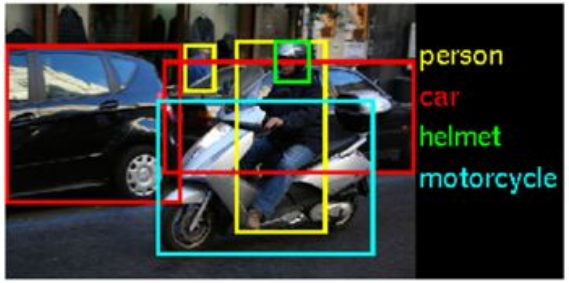
In 2010 teams had
75-50% error

In 2011 teams had
75-25% error

The ImageNet Challenge and the birth of CNNs



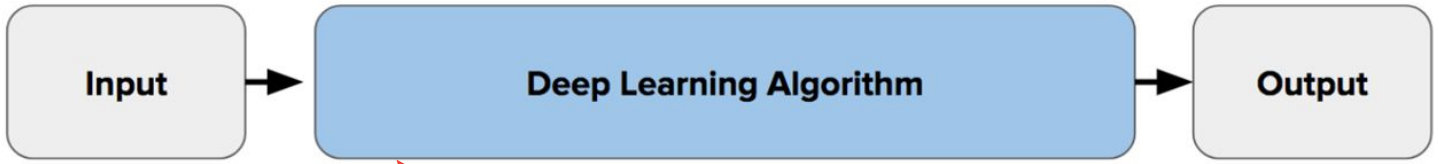




In 2012 still no team had less than 25% error barrier except **AlexNet at 15%**

The ImageNet Challenge and the birth of CNNs

Traditional Machine Learning Flow



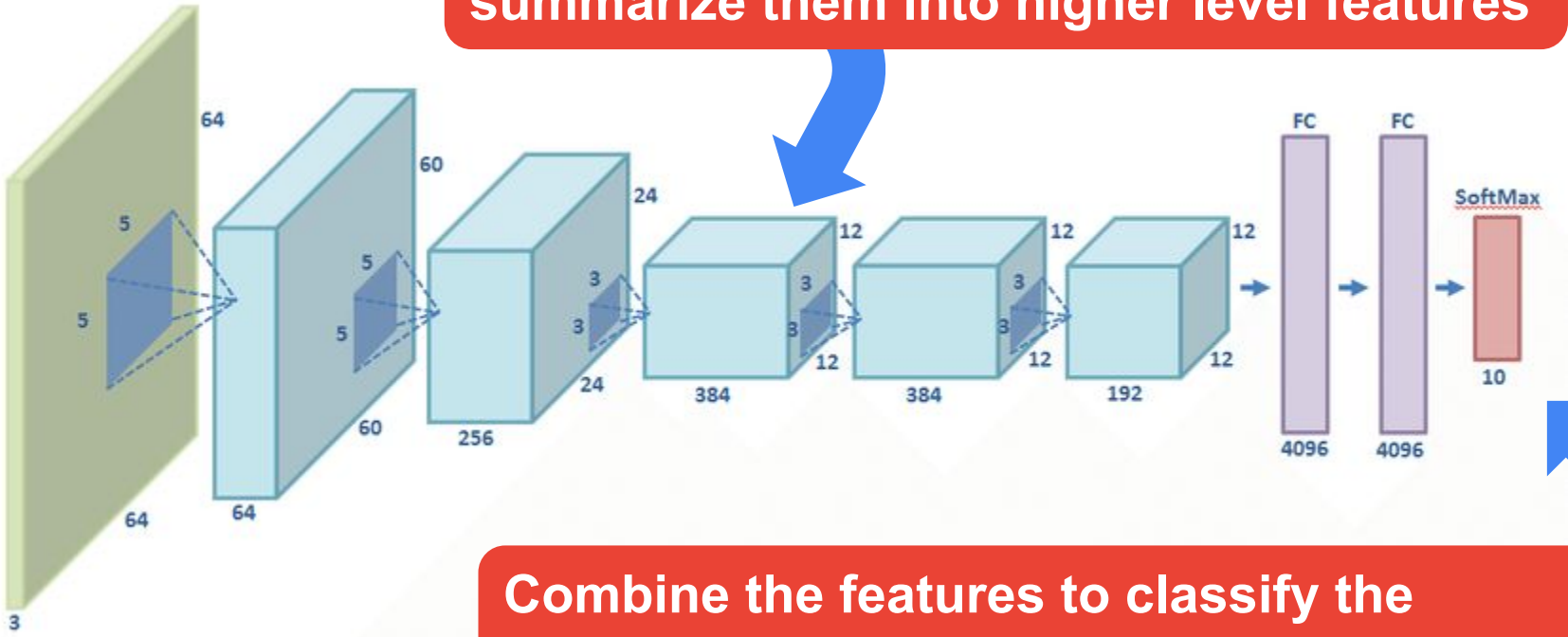
Deep Learning Flow

Let the computer figure out its own features and how to combine them!

AlexNet Paper

AlexNet

Use convolutions to find features and the summarize them into higher level features



Combine the features to classify the various objects in the dataset

Colab Link

How might we find these features?

Convolutions

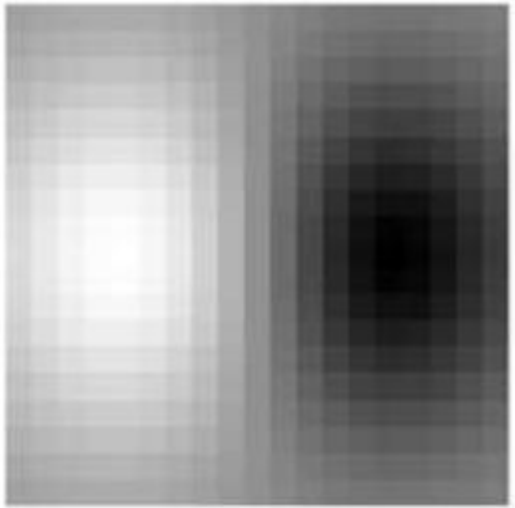
-1	0	1
-2	0	2
-1	0	1

Features

How might we find these features?

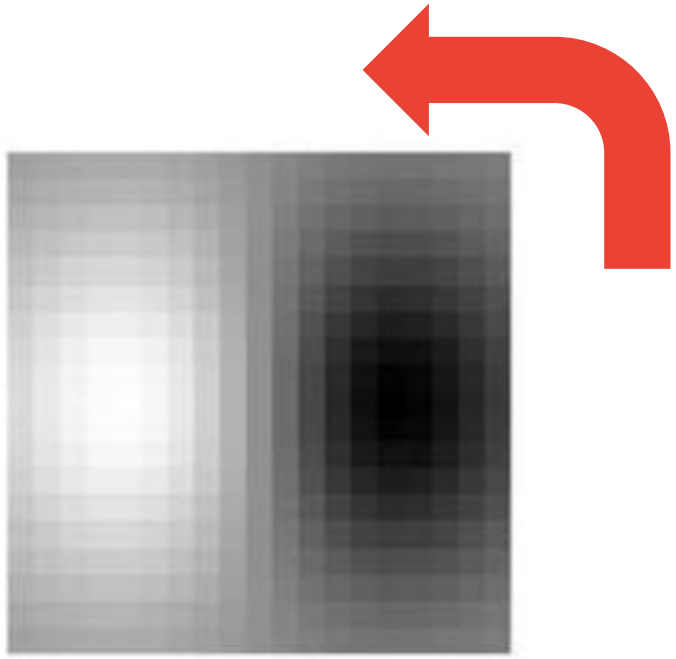
Convolutions

-1	0	1
-2	0	2
-1	0	1



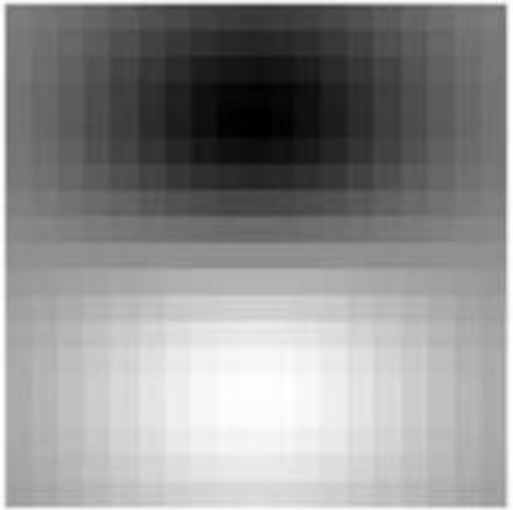
How might we find these features?

Convolutions



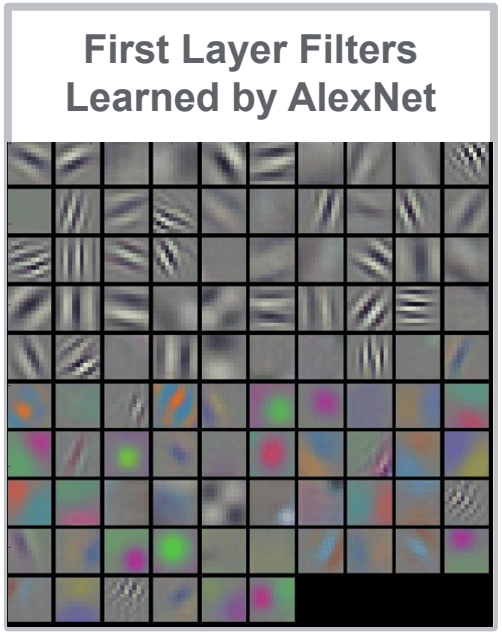
How might we find these features?

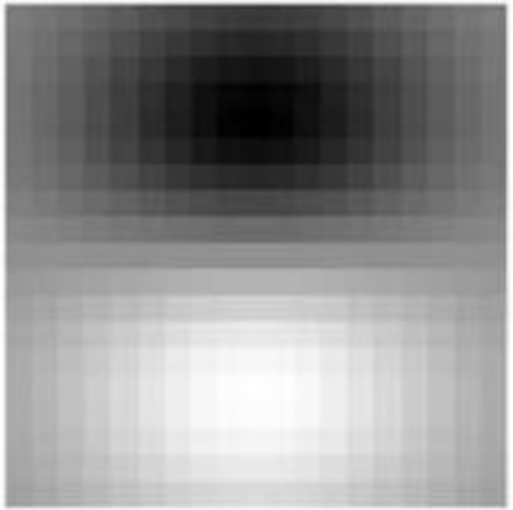
Convolutions



How might we find these features?

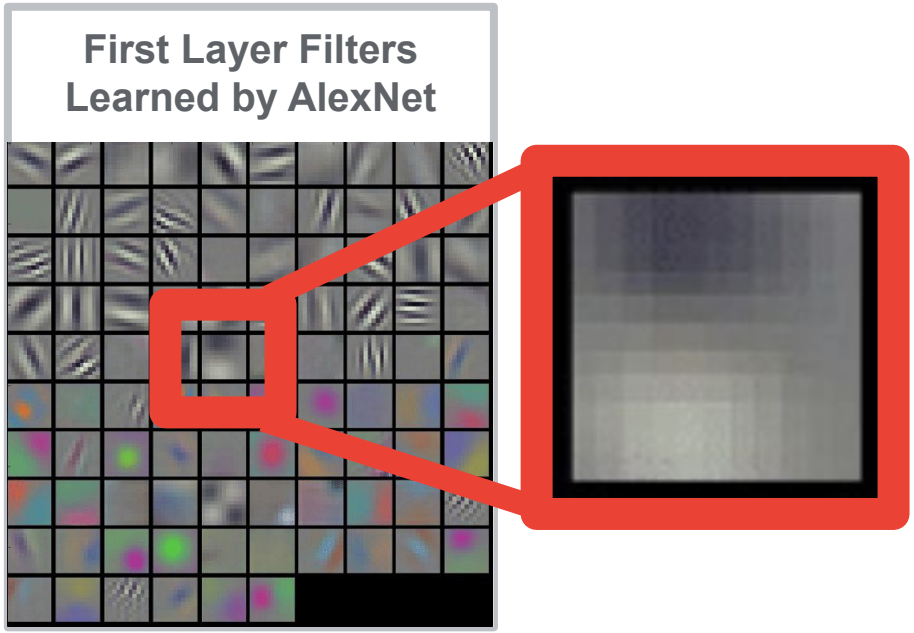
Convolutions

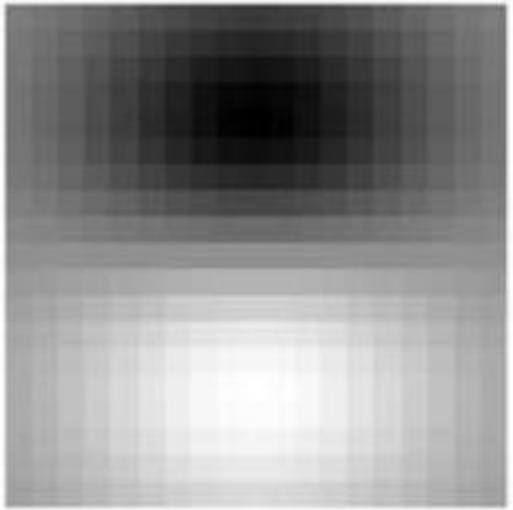




How might we find these features?

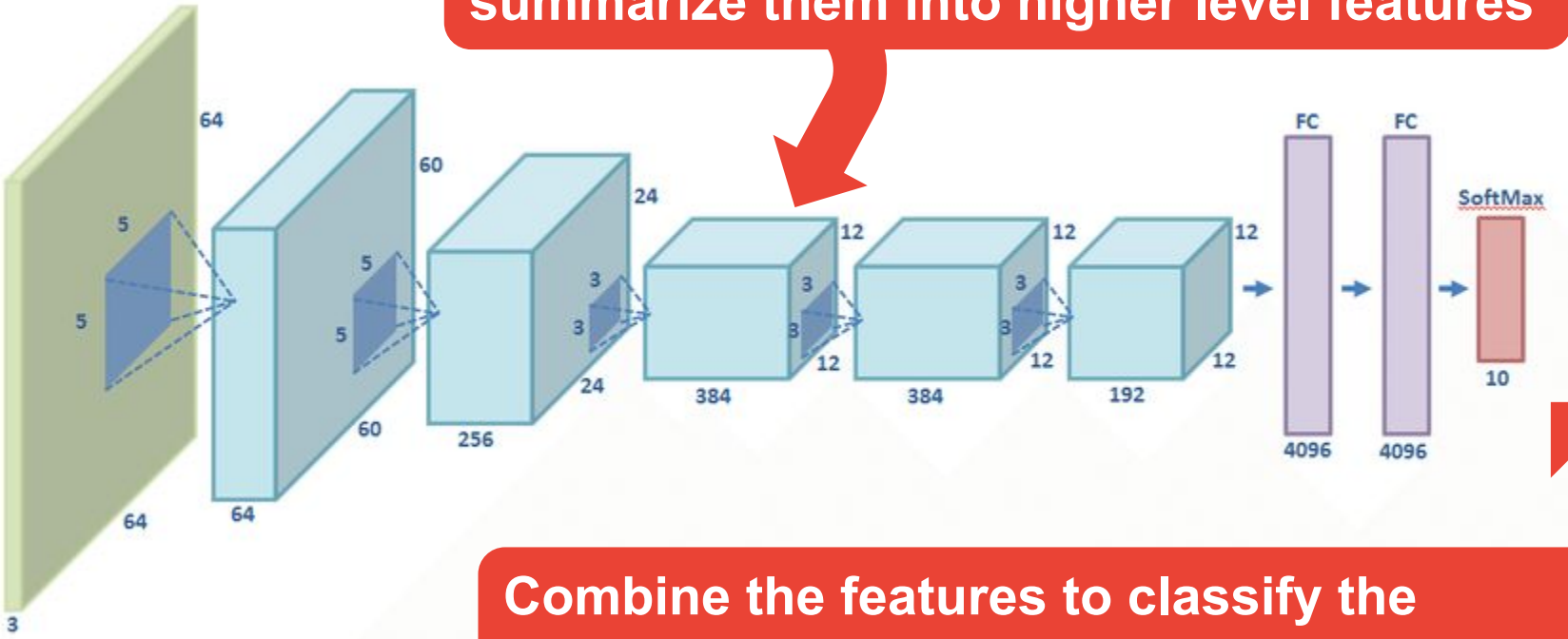
Convolutions



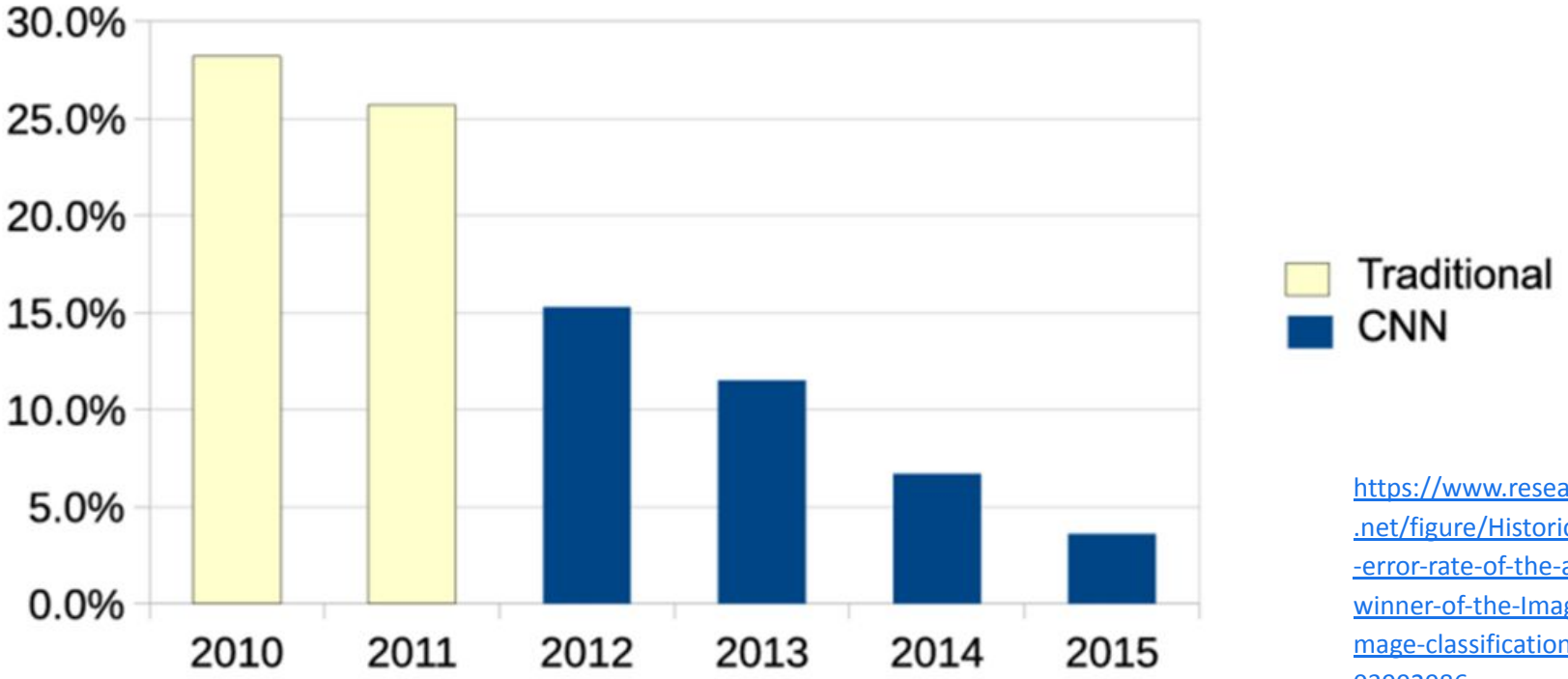


AlexNet Paper

AlexNet



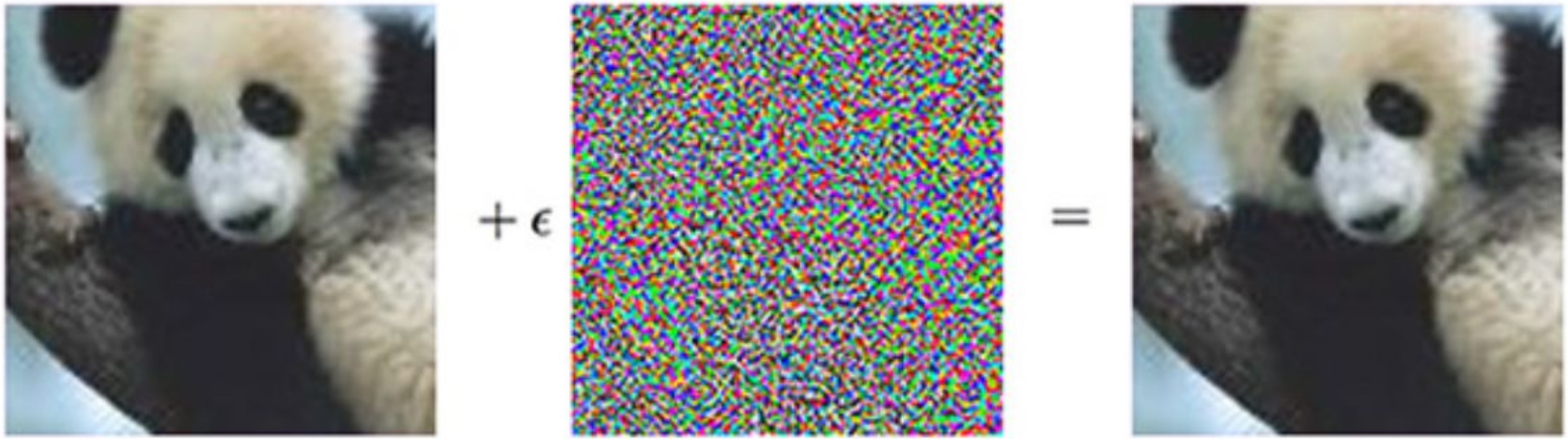
The ImageNet Challenge and the birth of CNNs



https://www.researchgate.net/figure/Historical-top5-error-rate-of-the-annual-winner-of-the-ImageNet-image-classification_fig7_303992986

A word of caution...

Ackerman "Hacking the Brain With Adversarial Images"

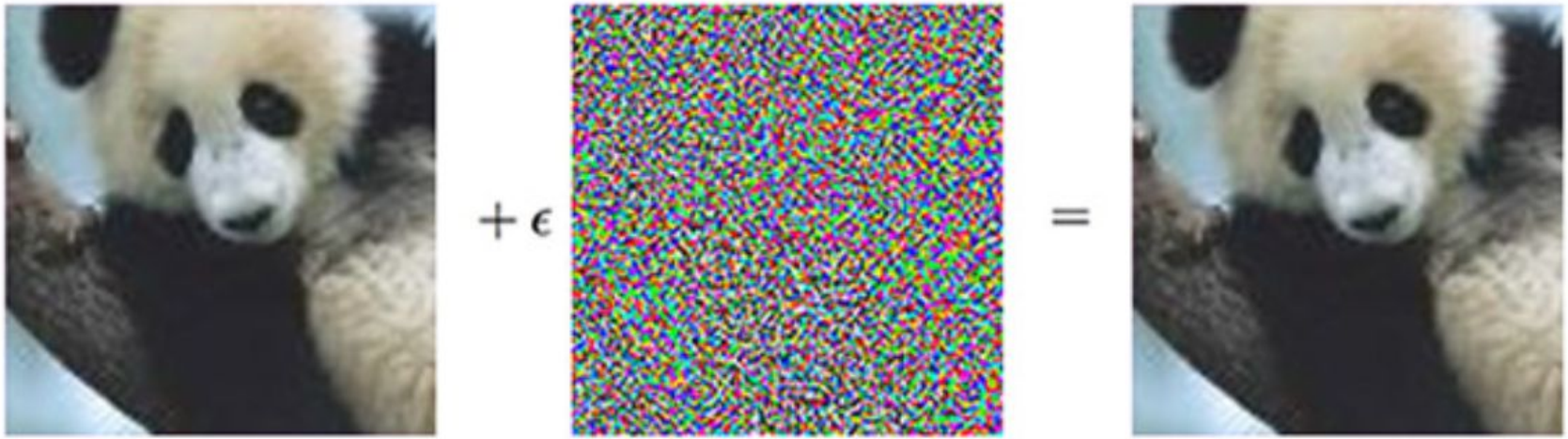


There is **no model** of the world semantically just mathematically

https://www.vox.com/future-perfect/2019/4/8/18297410/ai-tesla-self-driving-cars-adversarial-machine-learning

A word of caution...

Ackerman "Hacking the Brain With Adversarial Images"



There is **no model** of the world semantically just mathematically

Today's Agenda

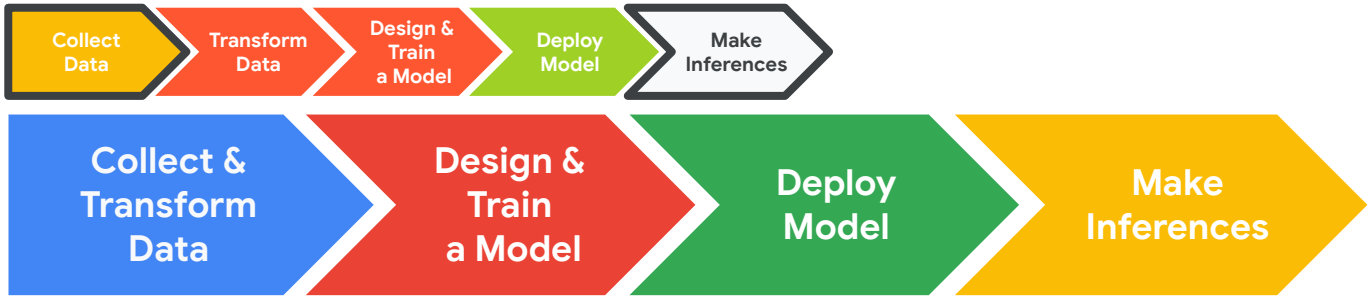
- Preprocessing for Keyword Spotting
- **Convolutional Neural Networks for Image Classification**
- Hands-on: KWS Data Collection with Edge Impulse
- Hands-on: Training our Model with Edge Impulse
- Hands-on: Testing our Model in the Real World
- Summary

Today's Agenda

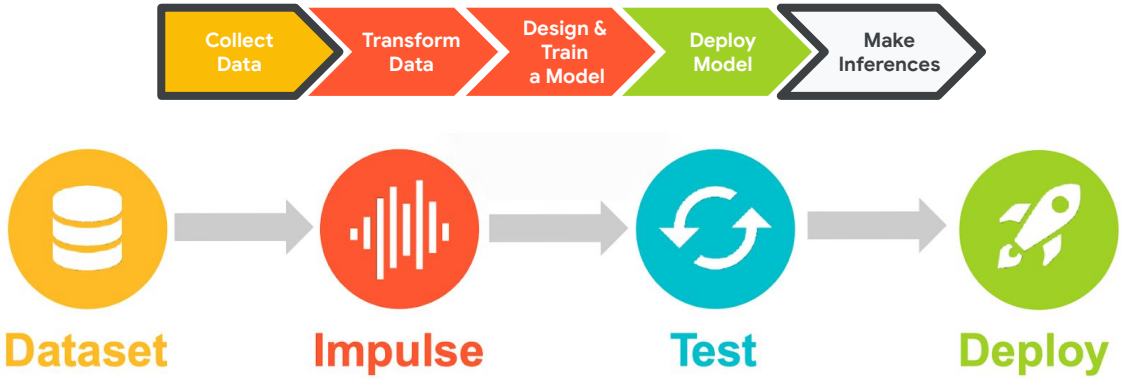
- Preprocessing for Keyword Spotting
- Convolutional Neural Networks for Image Classification
- **Hands-on: KWS Data Collection with Edge Impulse**
- Hands-on: Training our Model with Edge Impulse
- Hands-on: Testing our Model in the Real World
- Summary

Machine Learning **Workflow**

Machine Learning **Workflow**

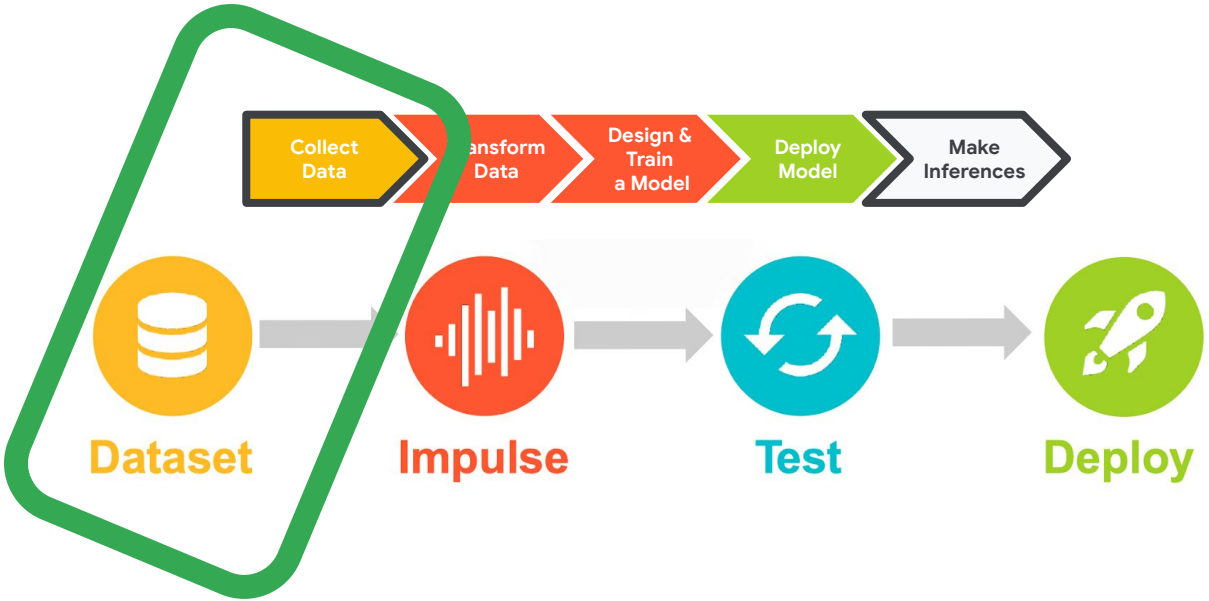


Edge Impulse Project Dashboard



- Dashboard
- Devices
- Data acquisition
- Impulse design
  - Create impulse
- EON Tuner
- Retrain model
- Live classification
- Model testing
- Versioning
- Deployment

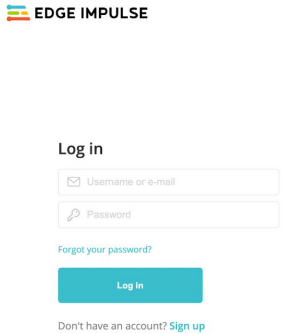
Edge Impulse Project Dashboard



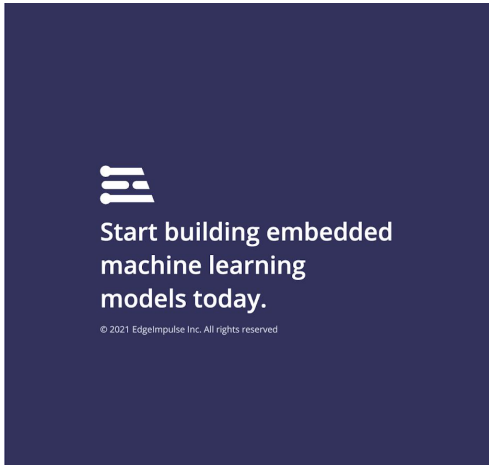
- Dashboard
- Devices
- Data acquisition
- Impulse design
  - Create impulse
- EON Tuner
- Retrain model
- Live classification
- Model testing
- Versioning
- Deployment

Create an Edge Impulse Account

1. Create an Edge Impulse account: https://studio.edgeimpulse.com/signup



2. Validate your email by clicking the link in the email sent to your account's email address



Activity: Create a Keyword Spotting Dataset

Collect ~30 samples each of the following classes of data:

- Keyword #1 Yá’át’ééh 👋
- Keyword #2 hágoónee’ 👋
- “Unknown” words that are not the keyword and background noise

https://docs.edgeimpulse.com/docs/pre-built-datasets/keyword-spotting

Activity: Create a Keyword Spotting Dataset

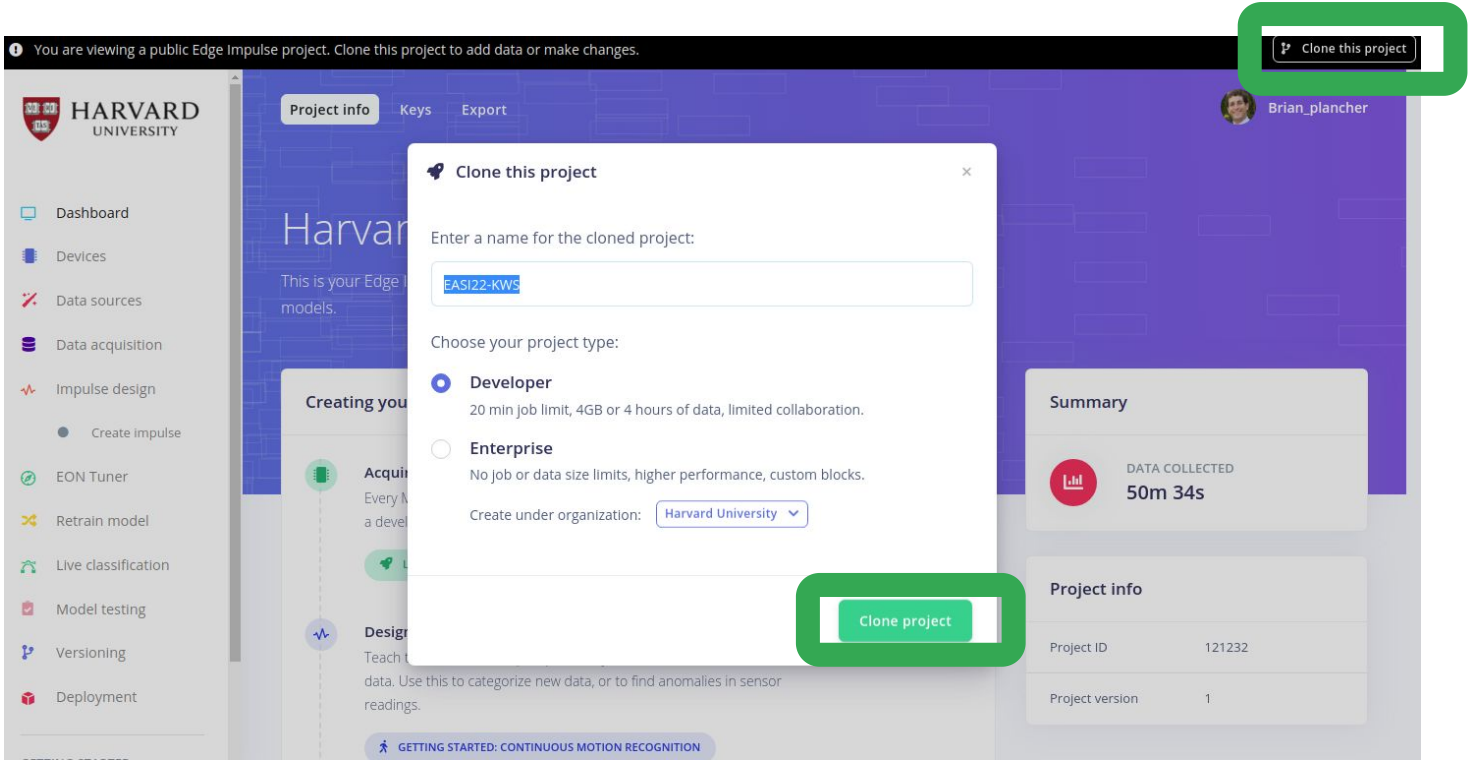
Collect ~30 samples each of the following classes of data:

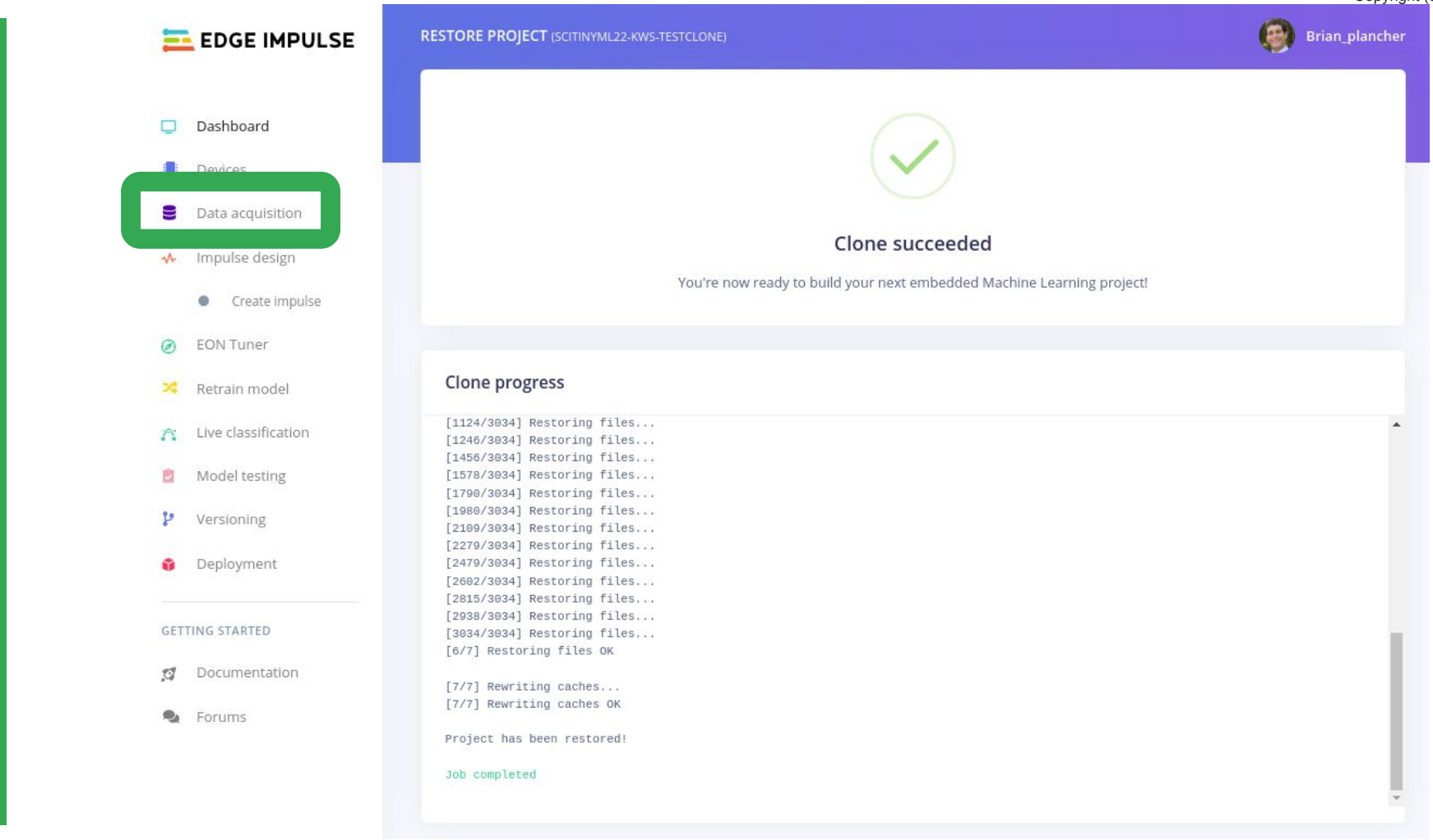
- Keyword #1 Yá’át’ééh 👋
- Keyword #2 hágoónee’ 👋
- ~~“Unknown” words that are not the and background noise~~

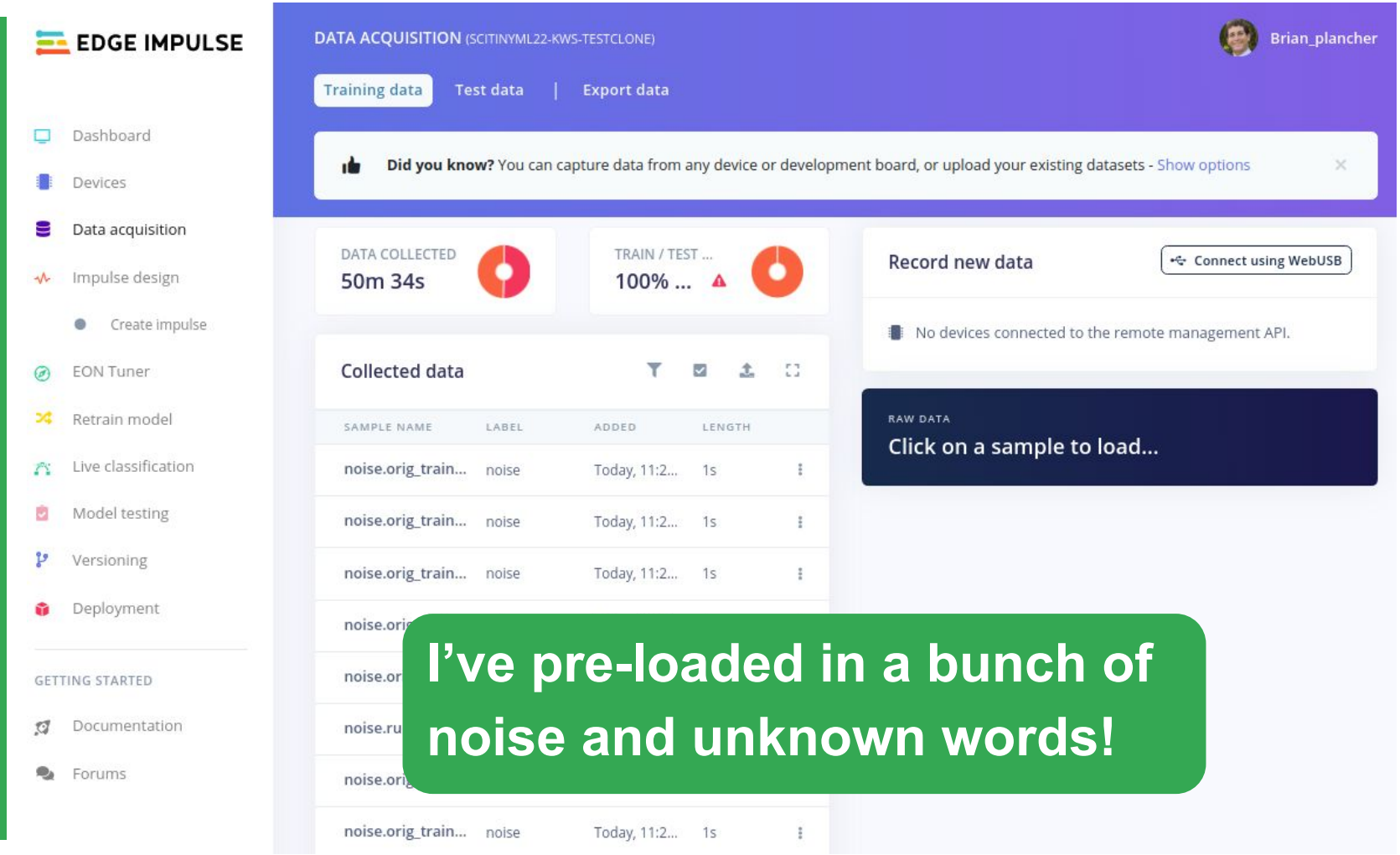
I’ve pre-loaded in a bunch of background noise and unknown words!

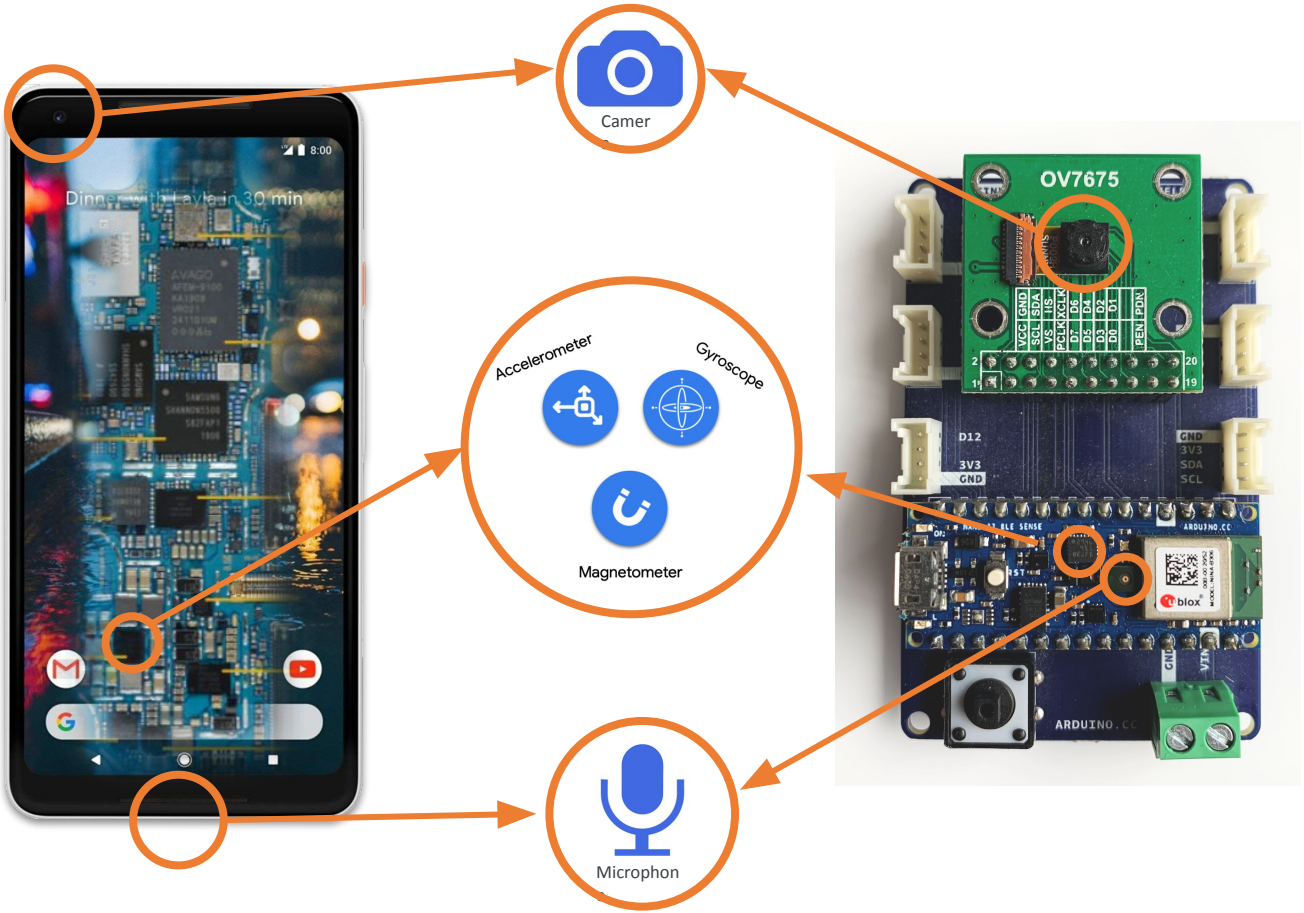
https://www.edgeimpulse.com/blog/public-projects-launch

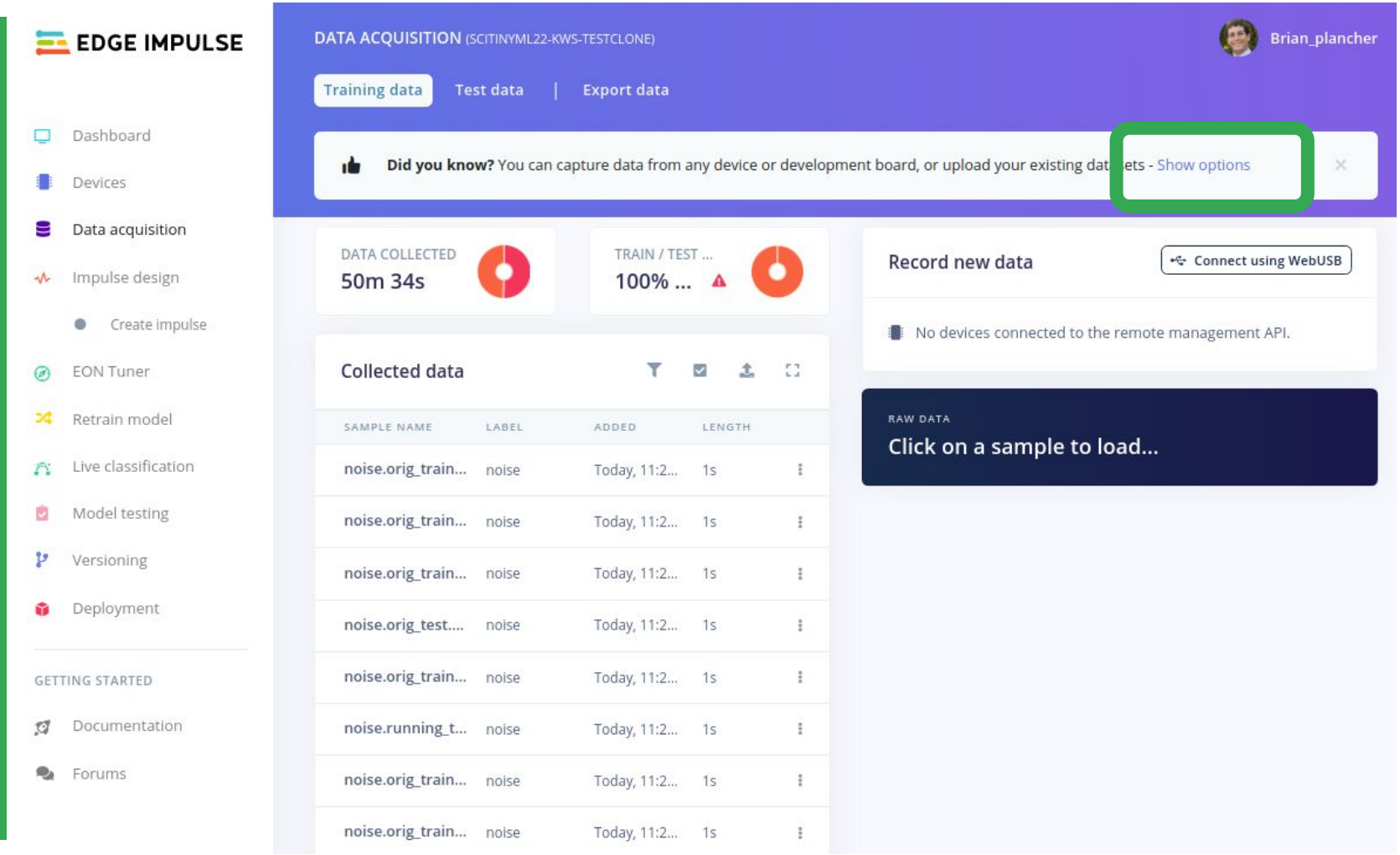
Clone my starter KWS project: https://bit.ly/EASI22-KWS

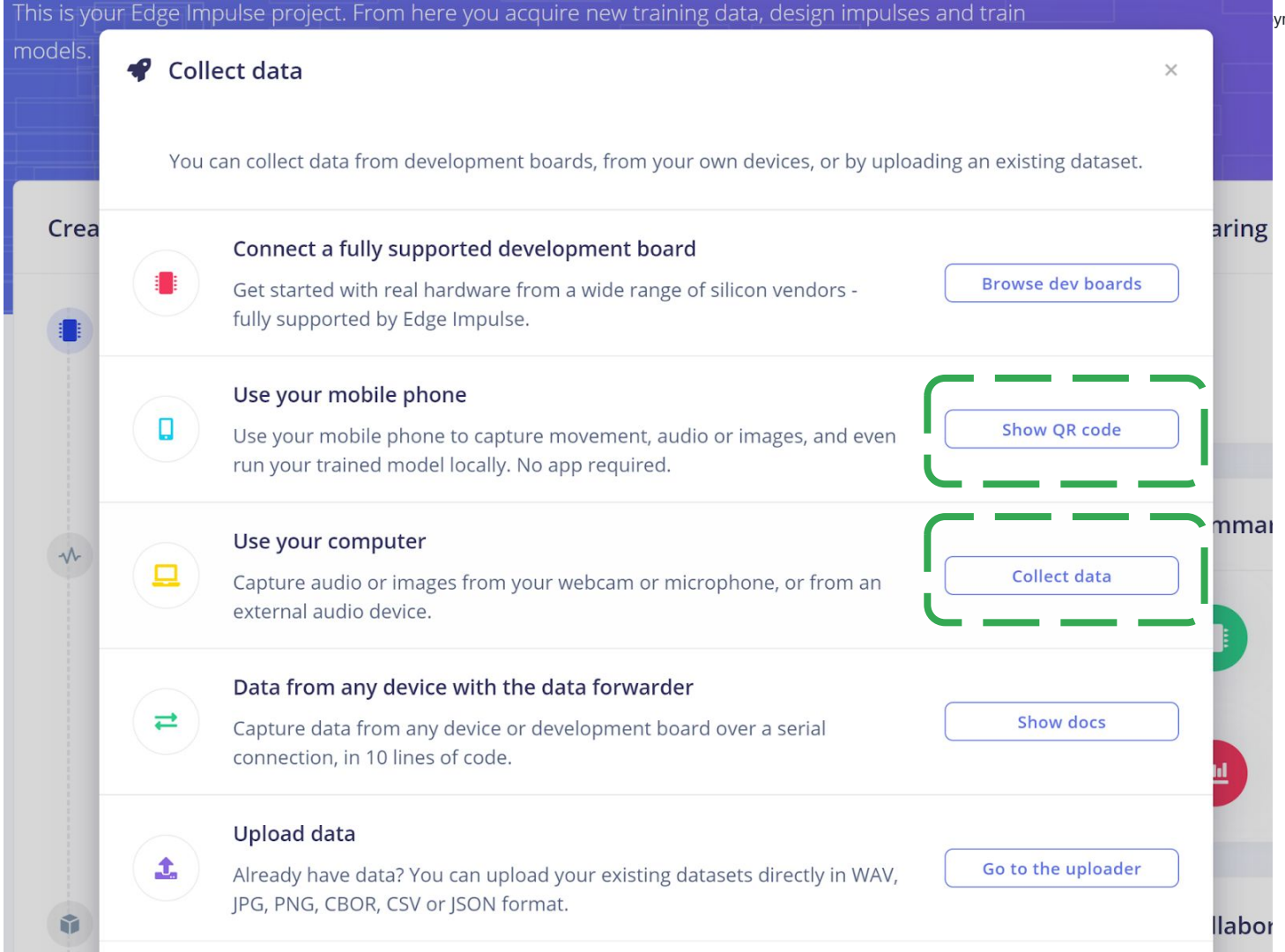


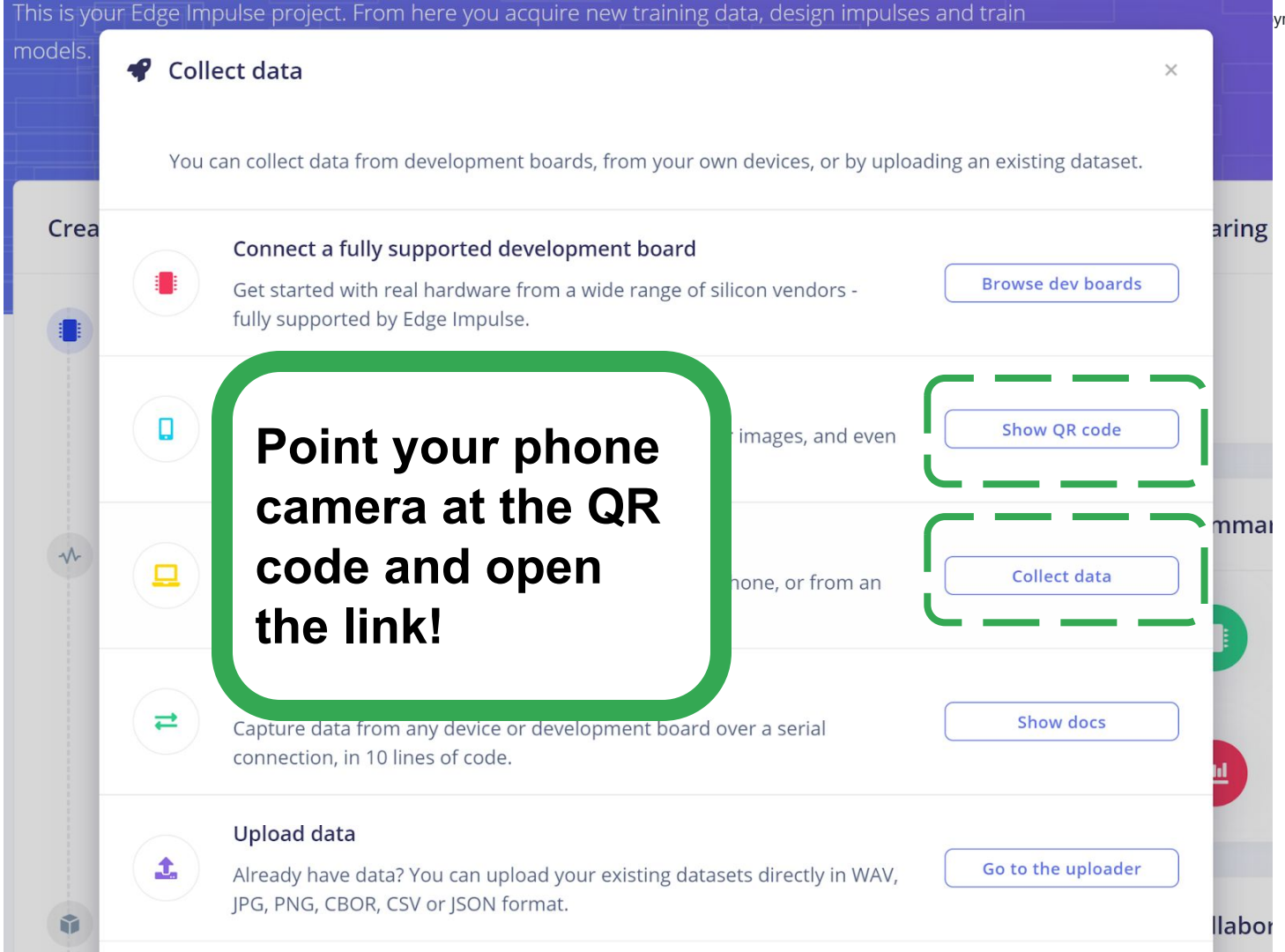


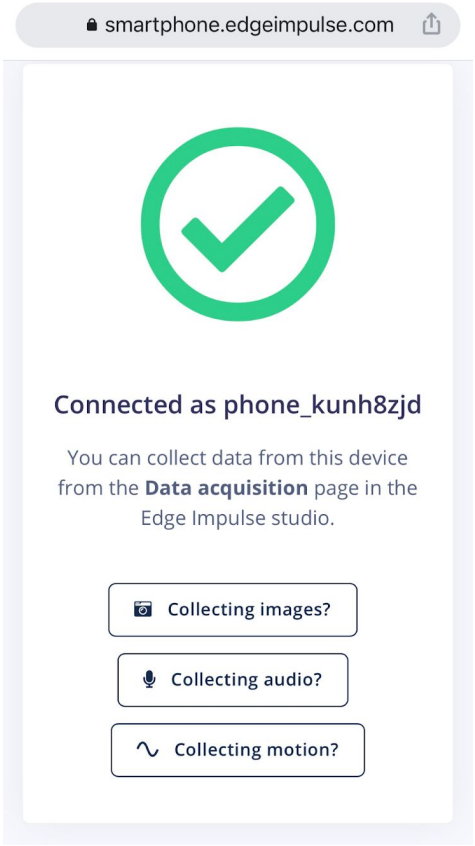


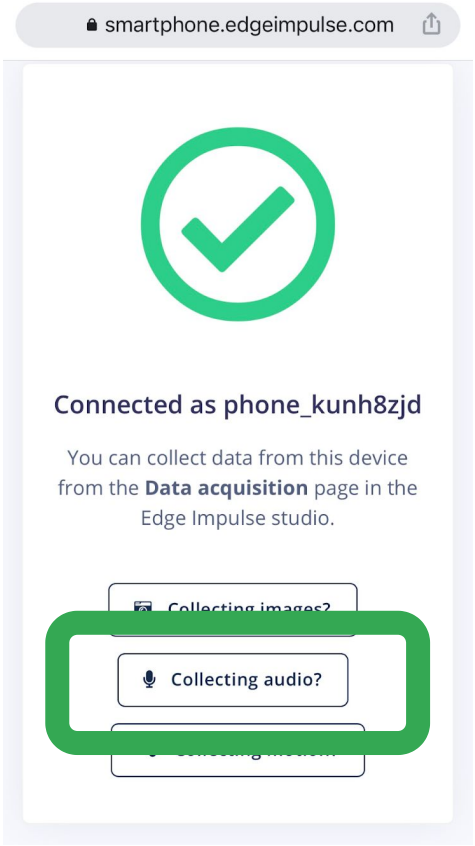


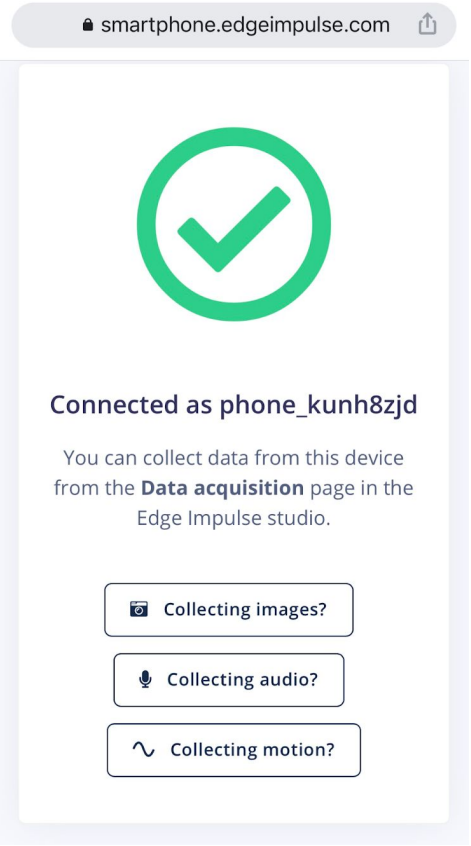


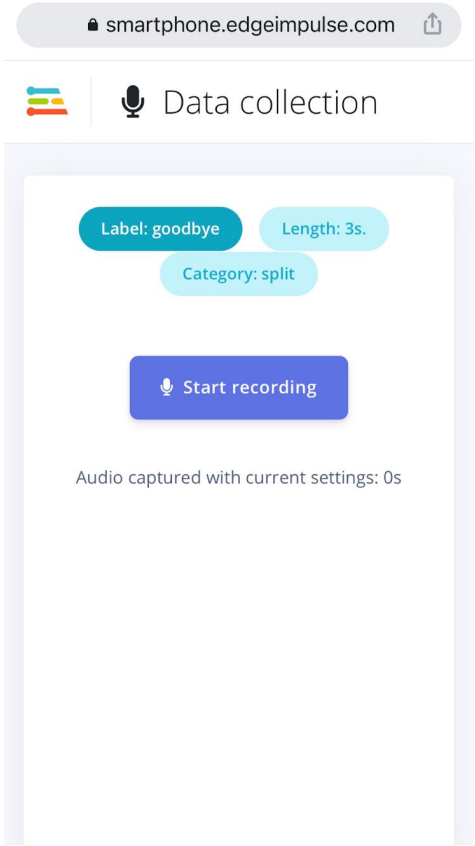


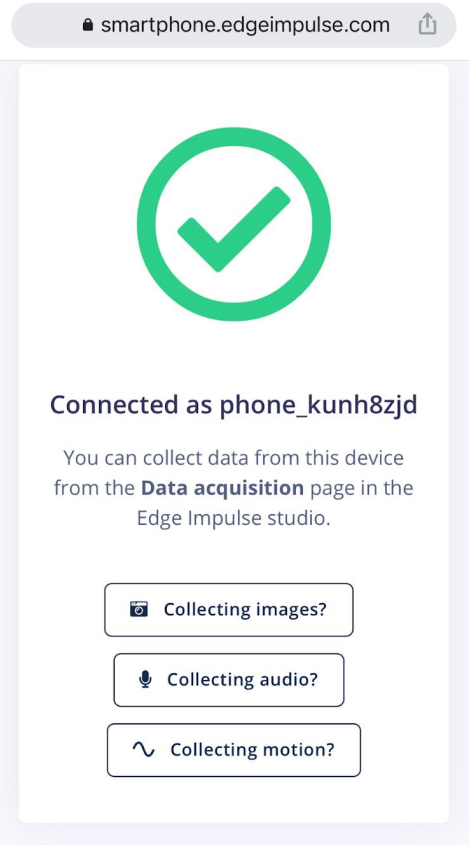


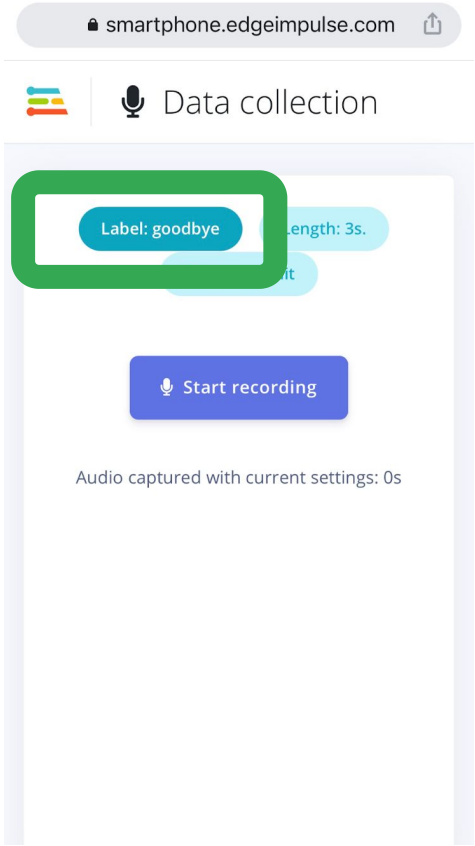


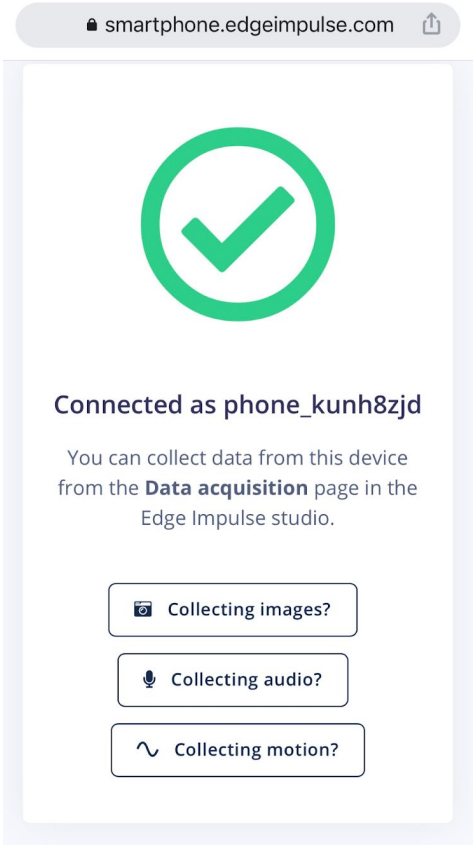


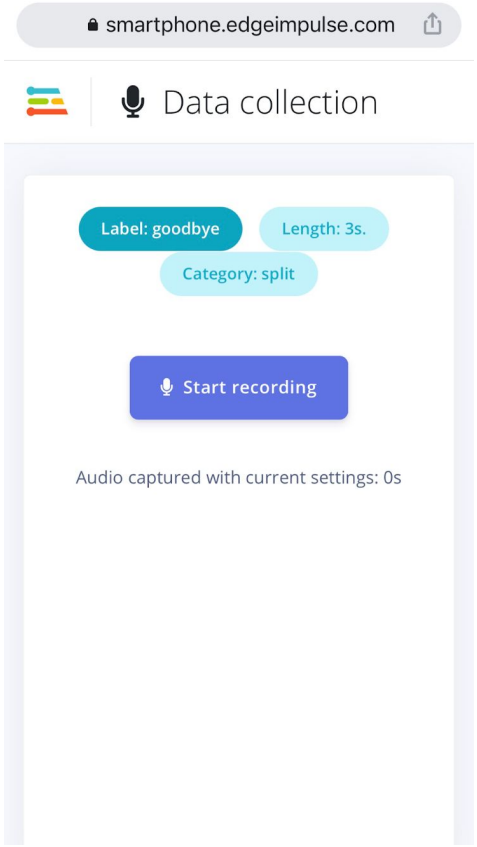


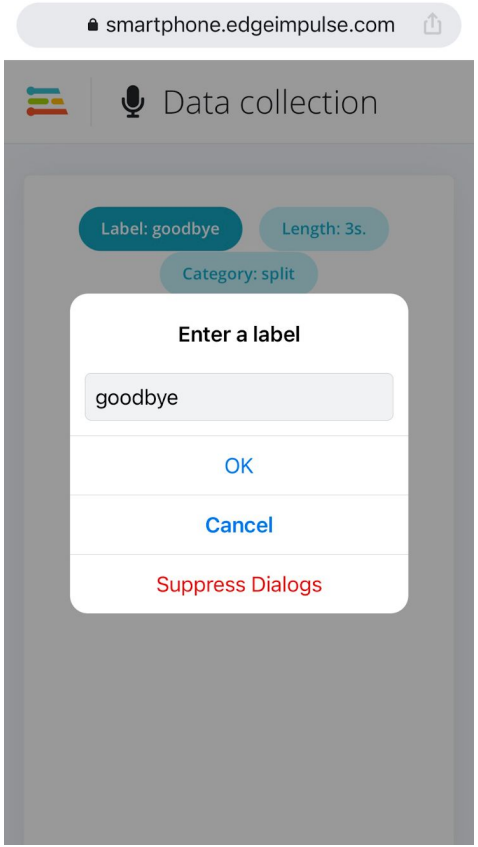


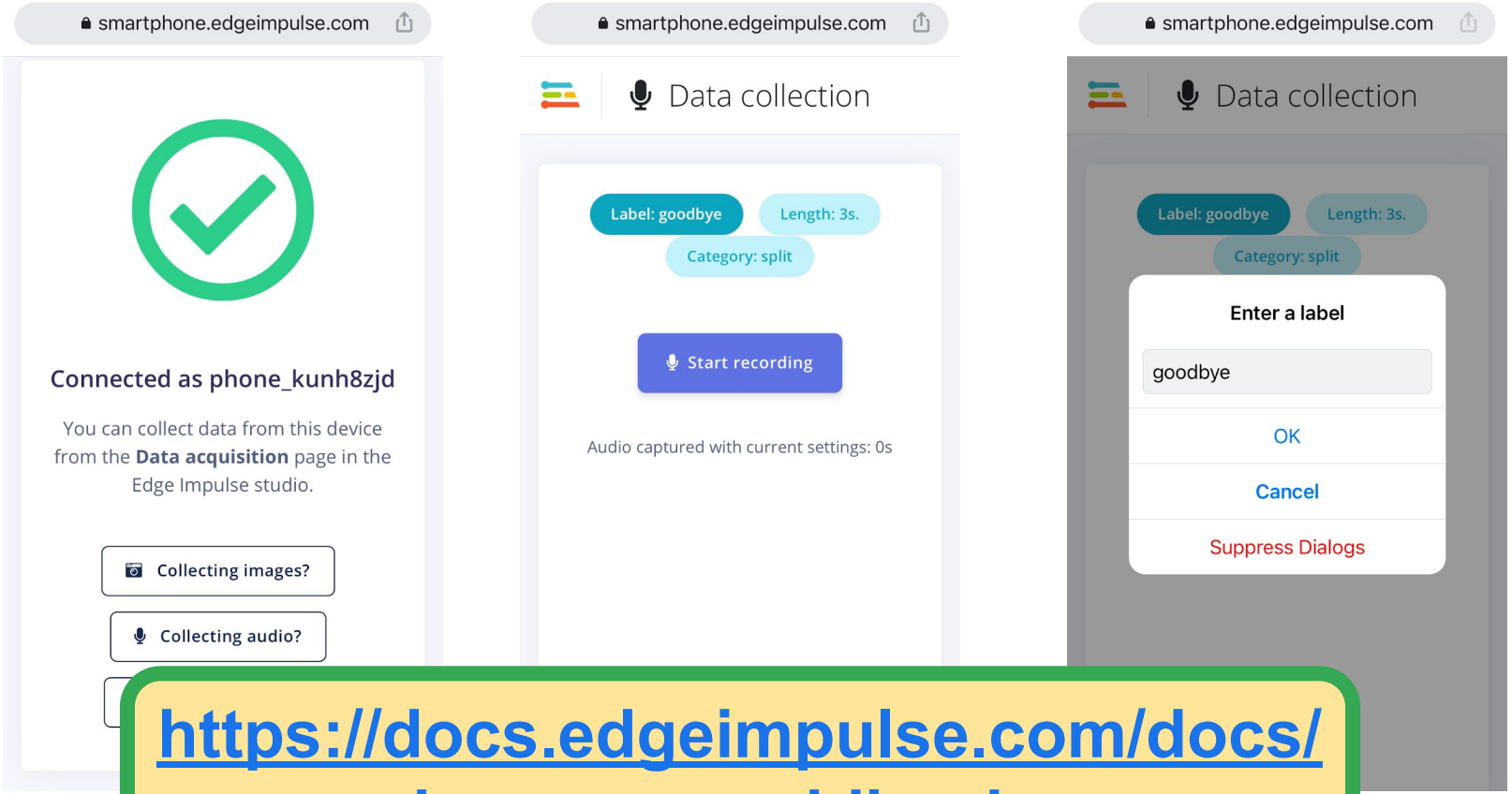












https://docs.edgeimpulse.com/docs/using-your-mobile-phone

DATA COLLECTED
50m 44s

TRAIN / TEST SPLIT
100% / 0%

Collected data

Delete selected (0) | Edit labels (0) | Move to test set (0) | Enable selected (0) | Disable selected (0)

SAMPLE NAME	LABEL	ADDED	LENGTH
yes.30u5okgq	yes	Today, 14:24:58	10s
noise.orig_train.Hallway_1.wav.7...	noise	Today, 11:22:57	1s
noise.orig_train.Metro_1.wav.297...	noise	Today, 11:22:57	1s
noise.orig_train.CafeTeria_1.wav....	noise	Today, 11:22:57	1s
noise.orig_test.Babble_4.wav.2000	noise	Today, 11:22:57	1s
noise.orig_train.AirportAnnounc...	noise	Today, 11:22:57	1s
noise.running_tap.wav.29000	noise	Today, 11:22:57	1s
noise.orig_train.Station_1.wav.20...	noise	Today, 11:22:57	1s
noise.orig_train.AirConditioner_9...	noise	Today, 11:22:57	1s
noise.orig_test.Typing_1.wav.160...	noise	Today, 11:22:57	1s
noise.orig_train.SqueakyChair_9....	noise	Today, 11:22:57	1s
noise.orig_train.AirportAnnounc...	noise	Today, 11:22:57	1s

1 2 3 4 5 6 7 ... 253

Record new data

Device: 6F:E3:4B:F3:11:23

Label: yes

Sample length (ms.): 10000

Sensor: Built-in microphone

Frequency: 16000Hz

Start sampling

RAW DATA
yes.30u5okgq

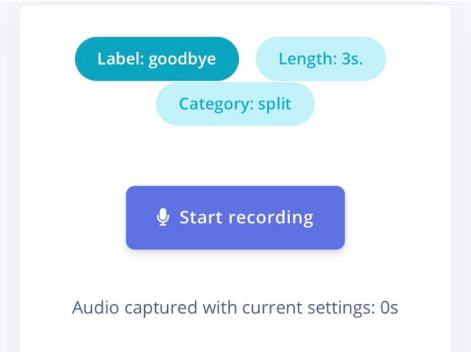
Activity: Create a Keyword Spotting Dataset

Collect **~30 samples each** of the following classes of data:

- **Keyword #1** Yá’át’ééh 👋 label: yaateeh
- **Keyword #2** hágoónee’ 👋 label: hagoonee

length: 10 seconds

We’ll resume in 10 minutes!



DATA COLLECTED
50m 44s

TRAIN / TEST SPLIT
100% / 0%

Collected data

Delete selected (0) | Edit labels (0) | Move to test set (0) | Enable selected (0) | Disable selected (0)

SAMPLE NAME	LABEL	ADDED	LENGTH
yes.30u5okgq	yes	Today, 14:24:58	10s
noise.orig_train.Hallway_1.wav.7...	noise	Today, 11:22:57	1s
noise.orig_train.Metro_1.wav.297...	noise	Today, 11:22:57	1s
noise.orig_train.CafeTeria_1.wav....	noise	Today, 11:22:57	1s
noise.orig_test.Babble_4.wav.2000	noise	Today, 11:22:57	1s
noise.orig_train.AirportAnnounc...	noise	Today, 11:22:57	1s
noise.running_tap.wav.29000	noise	Today, 11:22:57	1s
noise.orig_train.Station_1.wav.20...	noise	Today, 11:22:57	1s
noise.orig_train.AirConditioner_9...	noise	Today, 11:22:57	1s
noise.orig_test.Typing_1.wav.160...	noise	Today, 11:22:57	1s
noise.orig_train.SqueakyChair_9....	noise	Today, 11:22:57	1s
noise.orig_train.AirportAnnounc...	noise	Today, 11:22:57	1s

‹ 1 2 3 4 5 6 7 ··· 253 ›

Record new data

Device
6F:E3:4B:F3:11:23

Label
yes

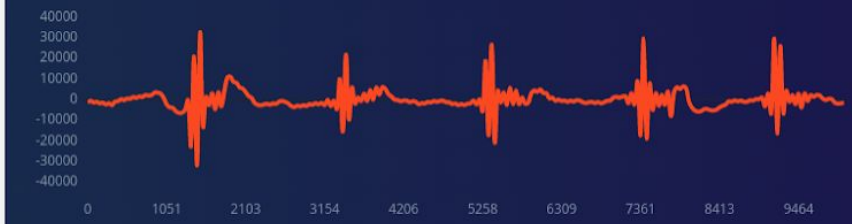
Sample length (ms.)
10000

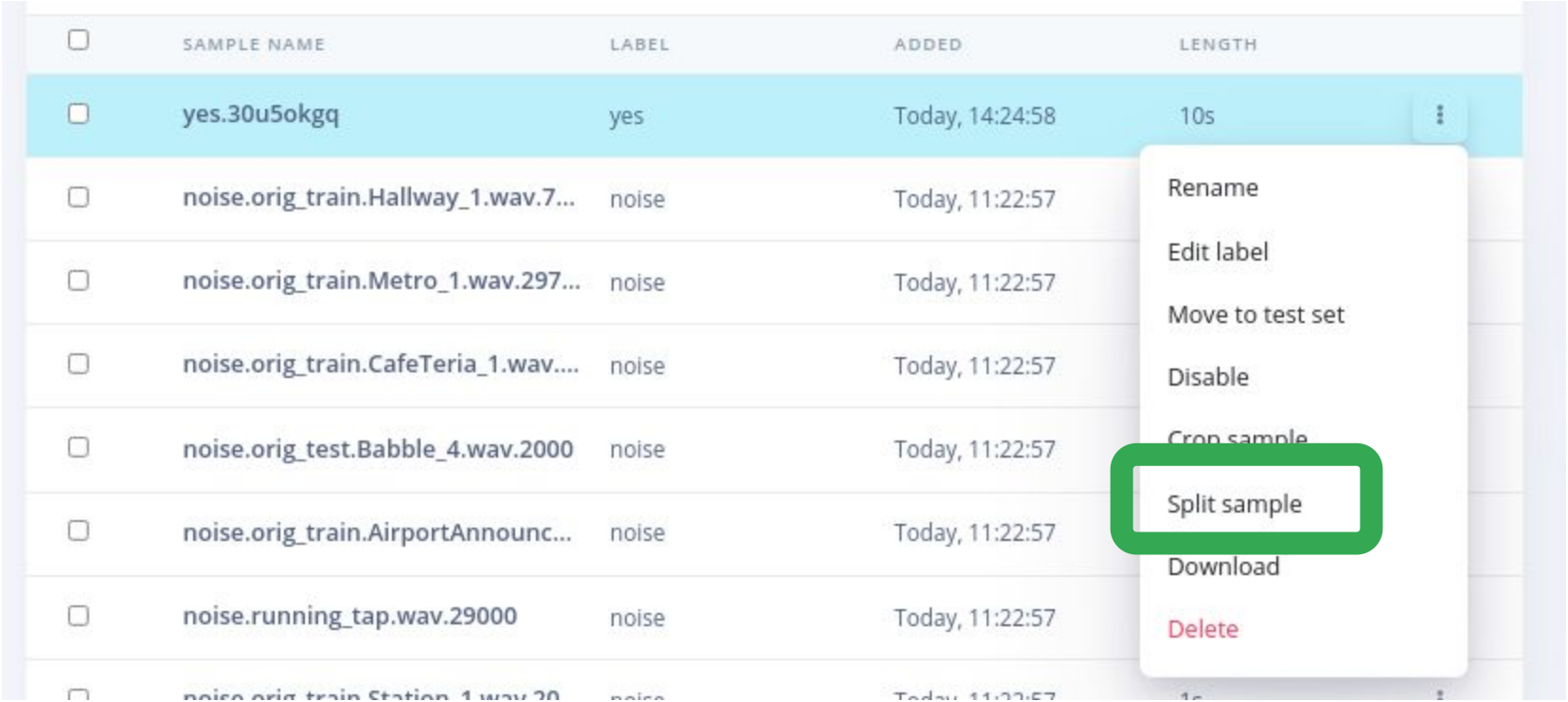
Sensor
Built-in microphone

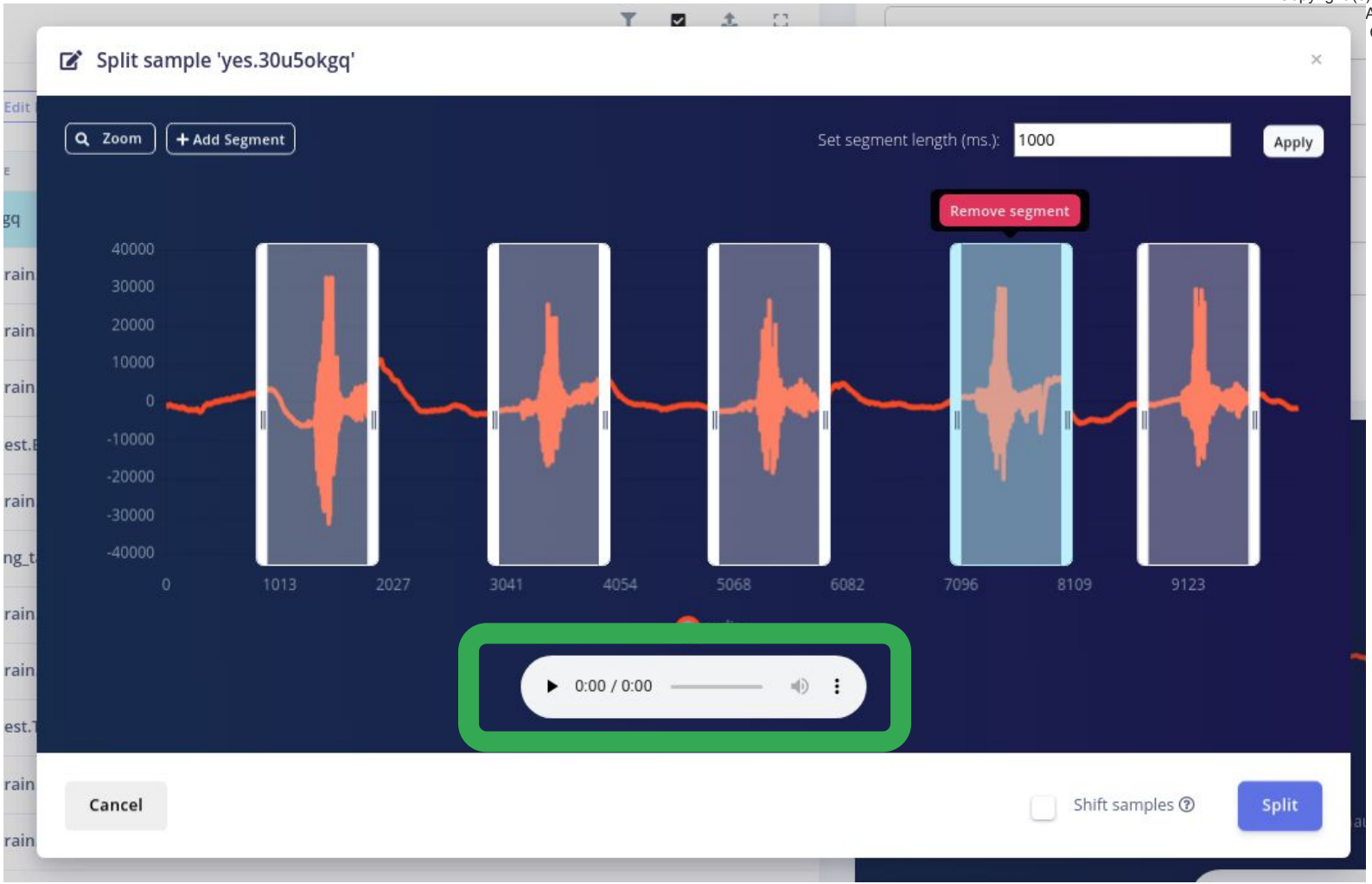
Frequency
16000Hz

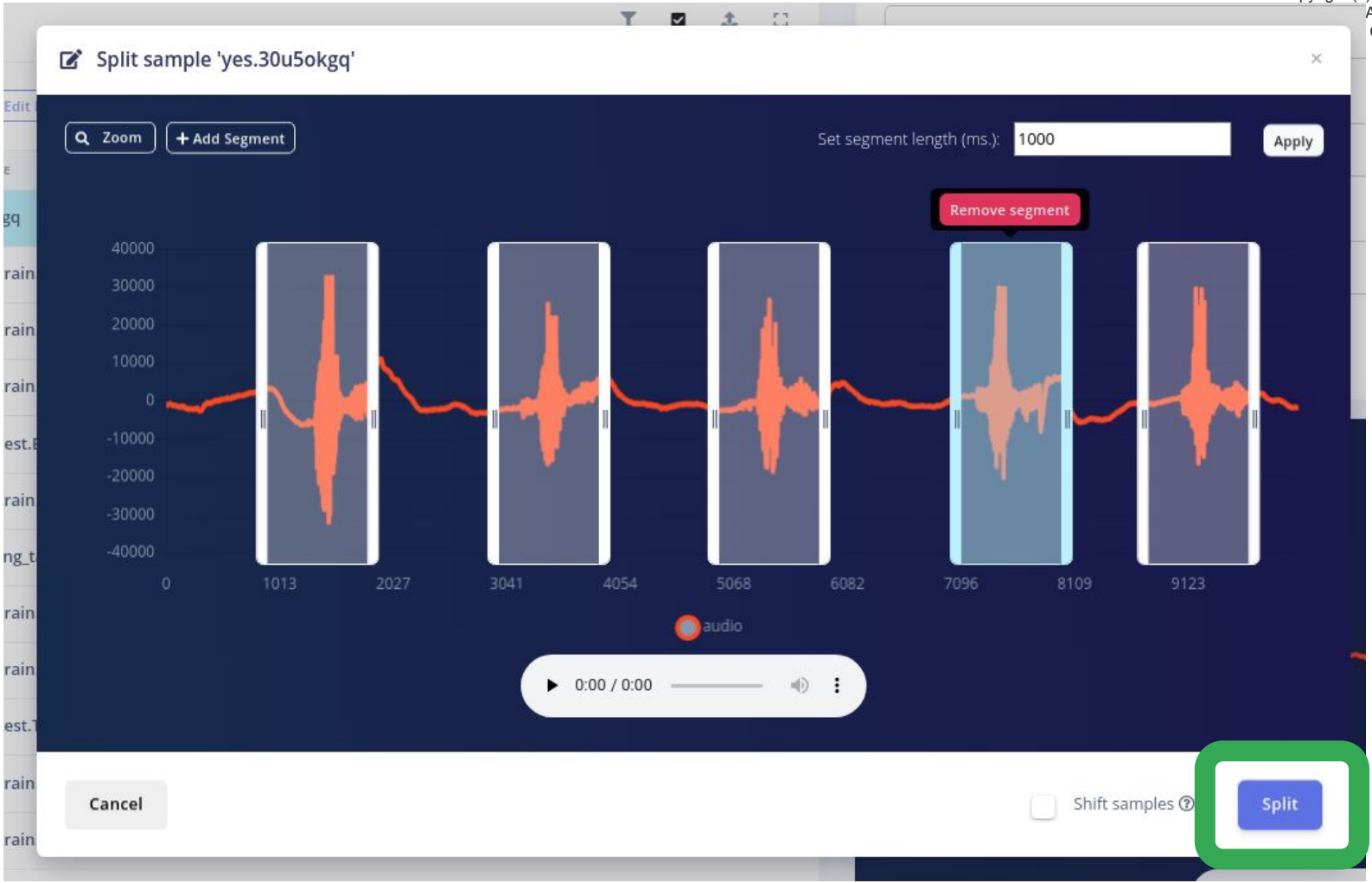
Start sampling

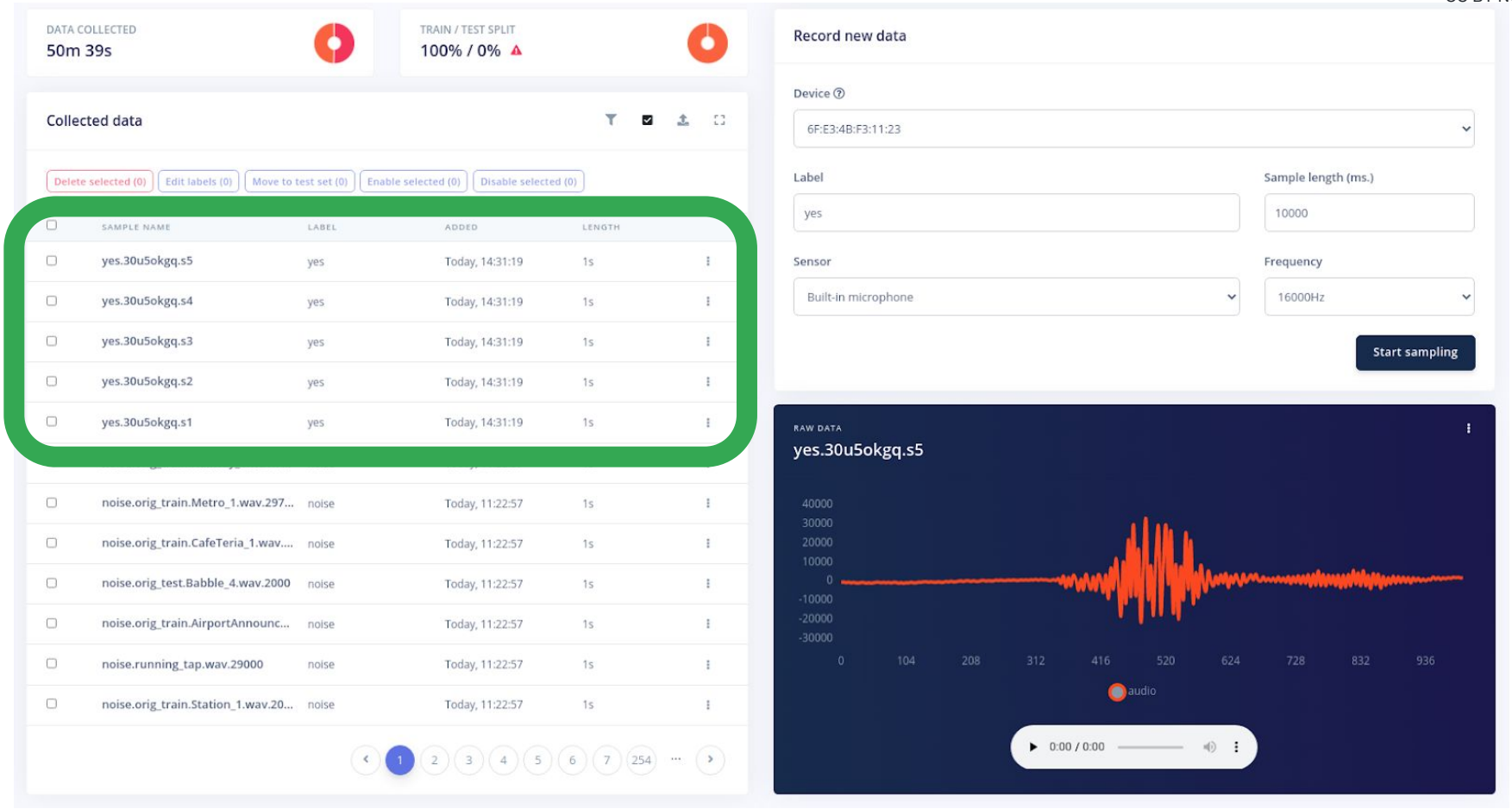
RAW DATA
yes.30u5okgq

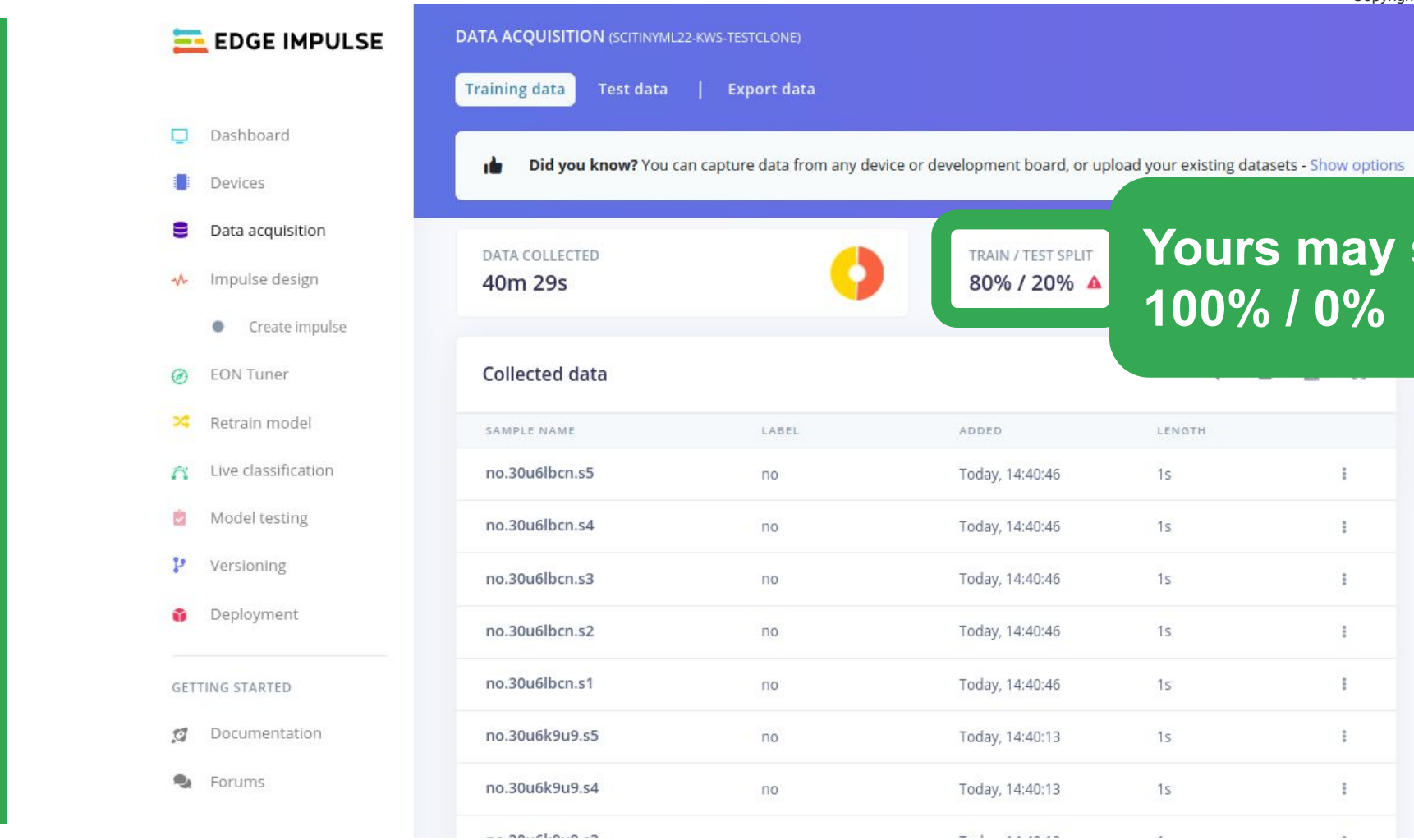


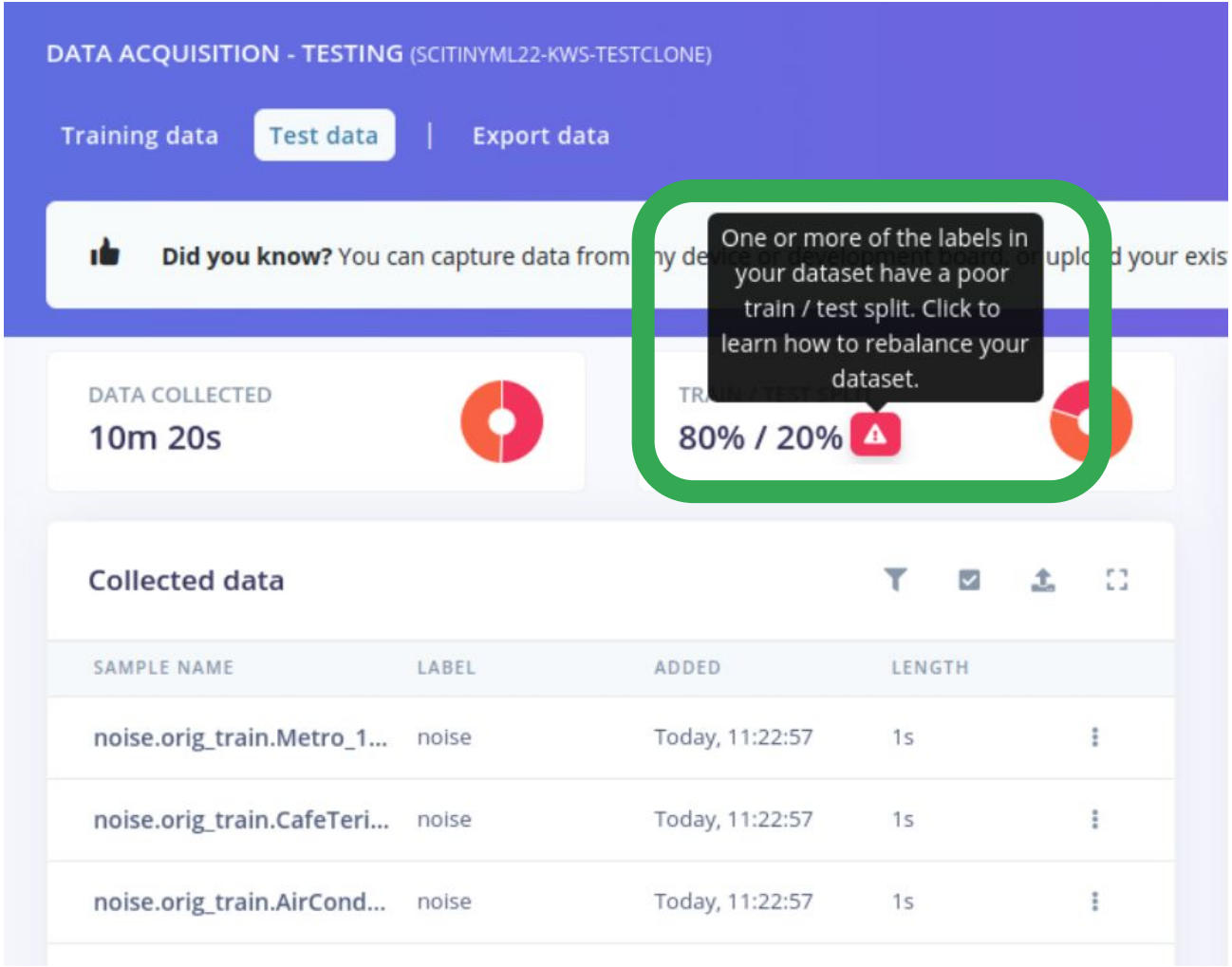


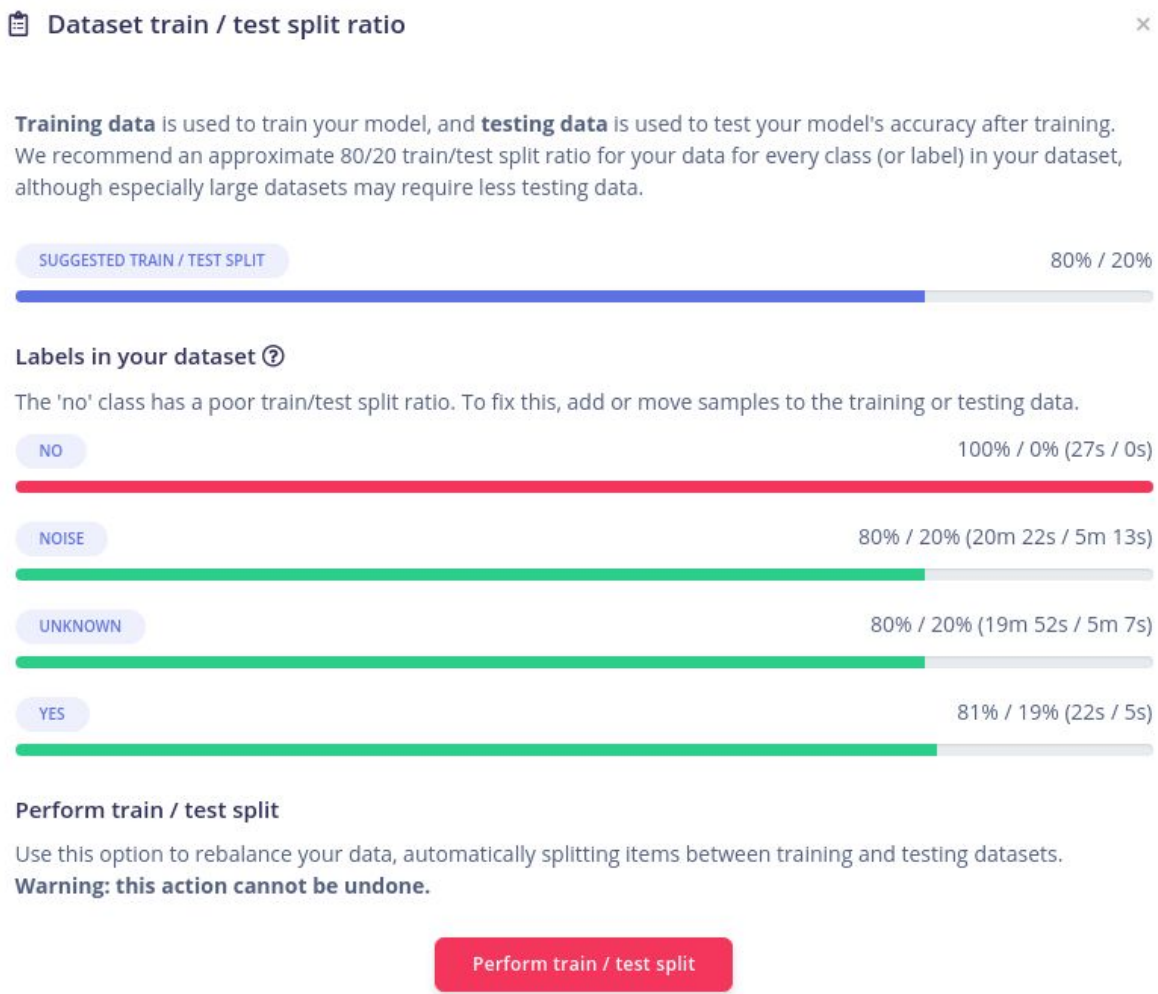


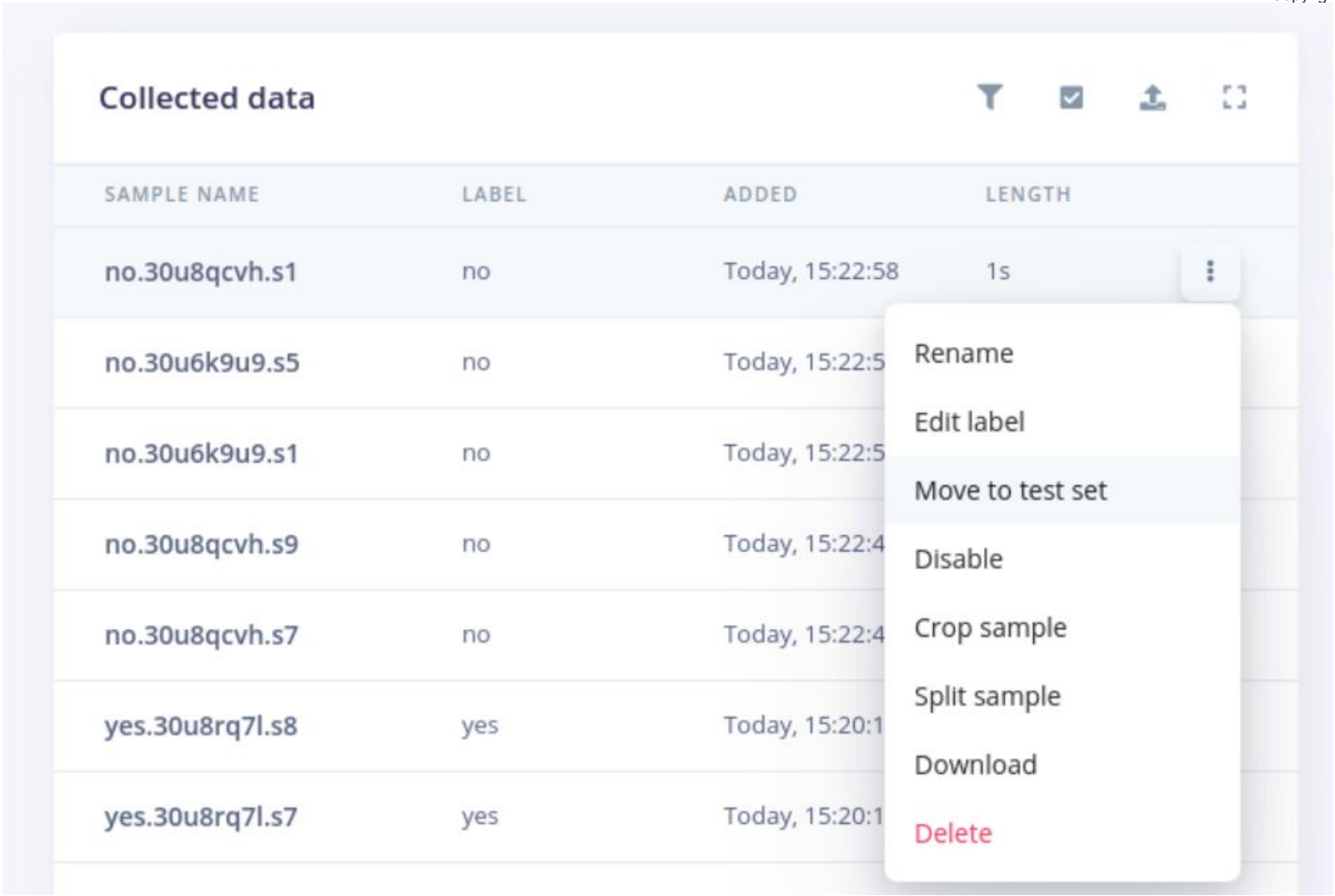


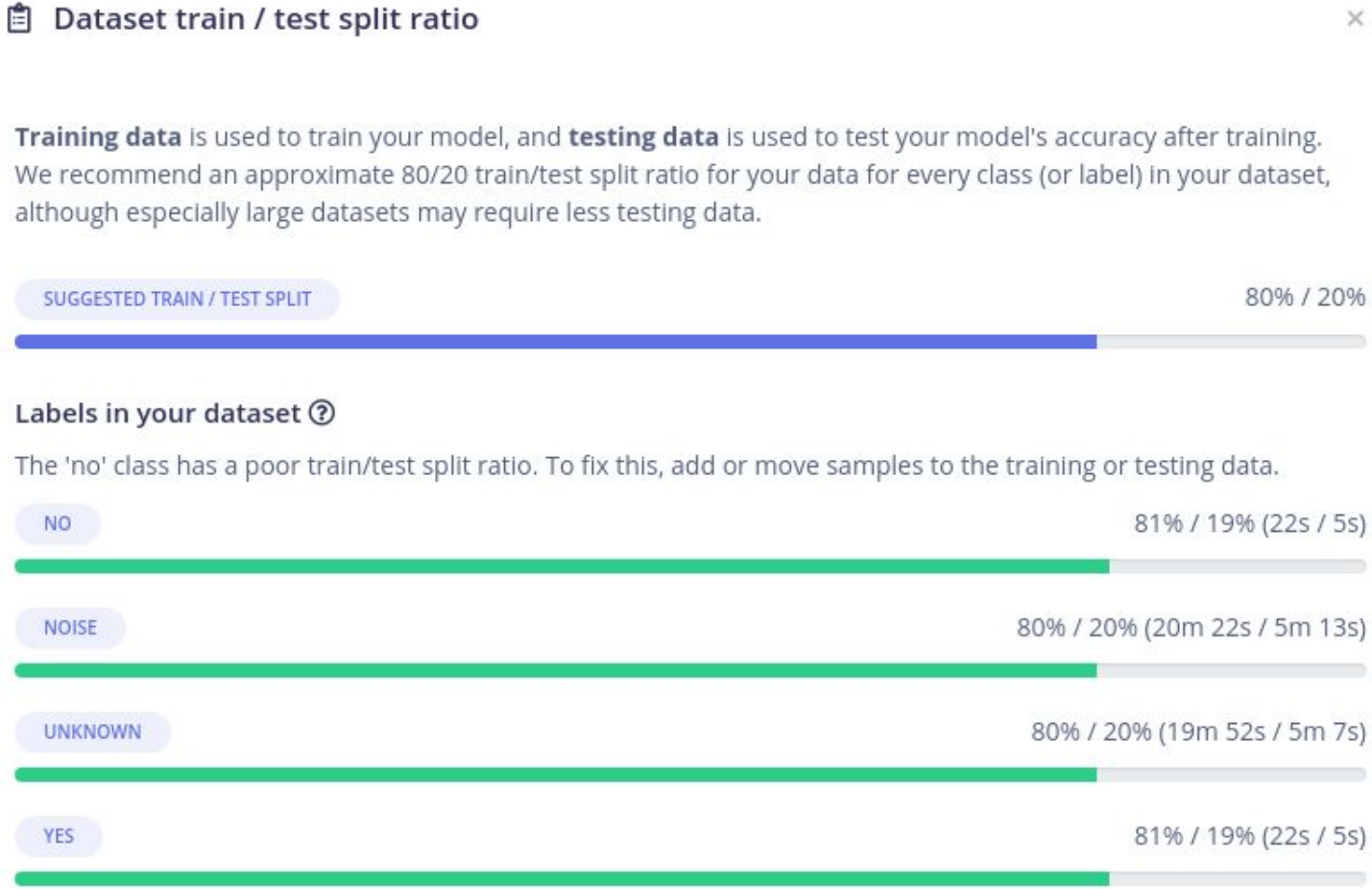












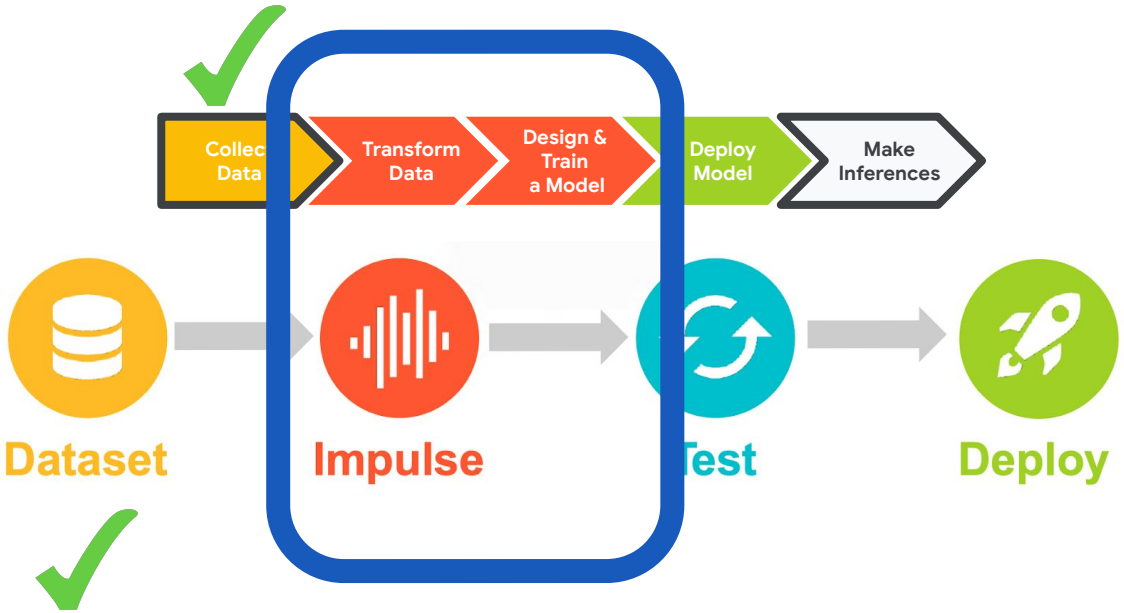
Today's Agenda

- Preprocessing for Keyword Spotting
- Convolutional Neural Networks for Image Classification
- **Hands-on: KWS Data Collection with Edge Impulse**
- Hands-on: Training our Model with Edge Impulse
- Hands-on: Testing our Model in the Real World
- Summary

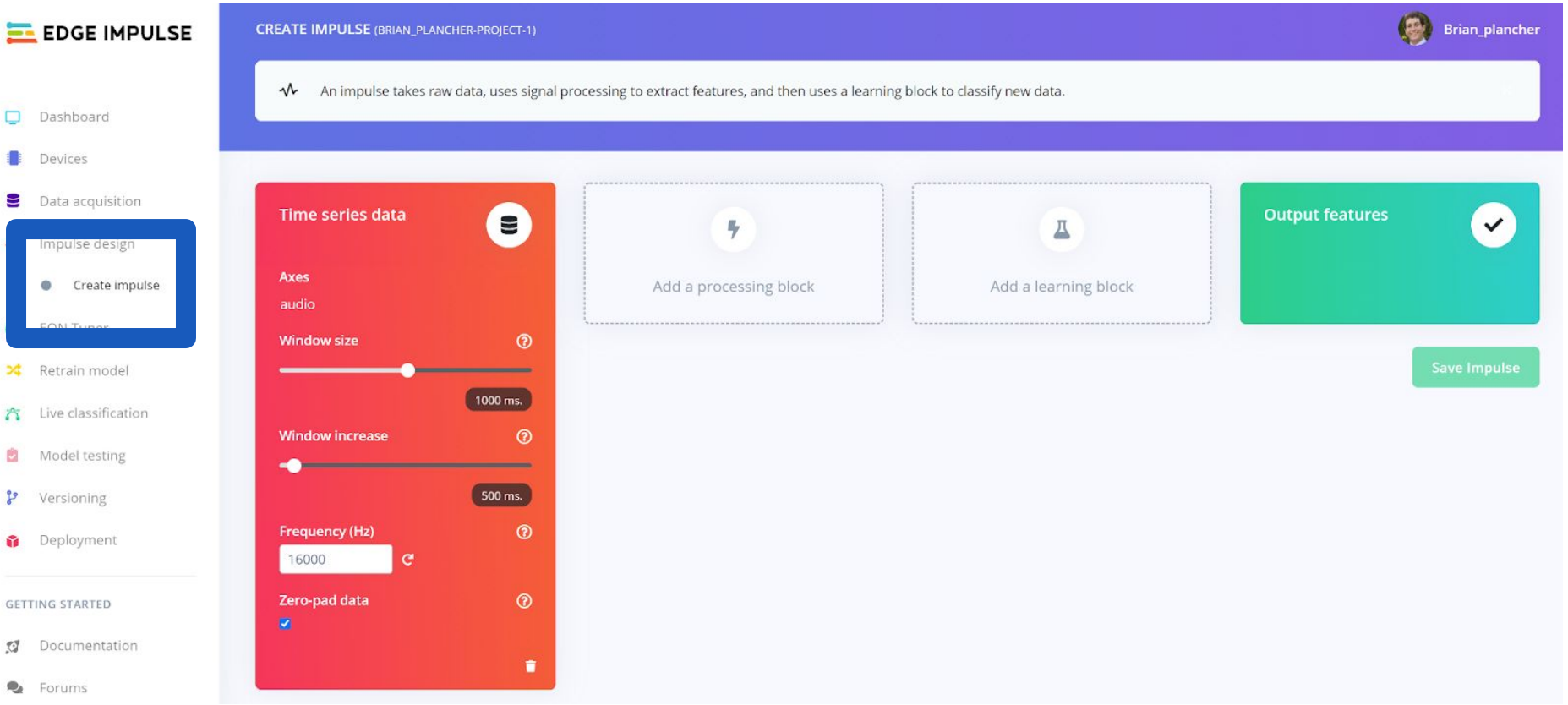
Today's Agenda

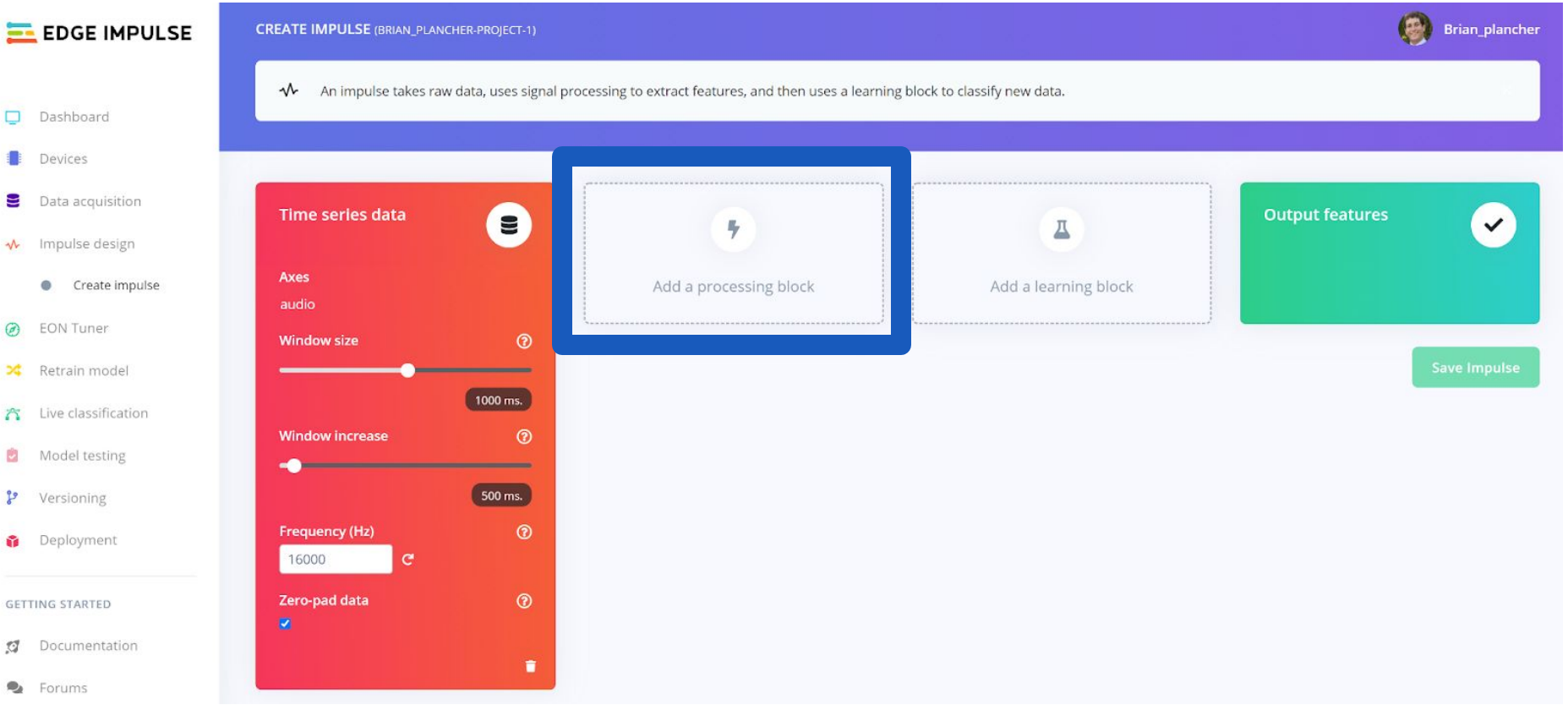
- Preprocessing for Keyword Spotting
- Convolutional Neural Networks for Image Classification
- Hands-on: KWS Data Collection with Edge Impulse
- **Hands-on: Training our Model with Edge Impulse**
- Hands-on: Testing our Model in the Real World
- Summary

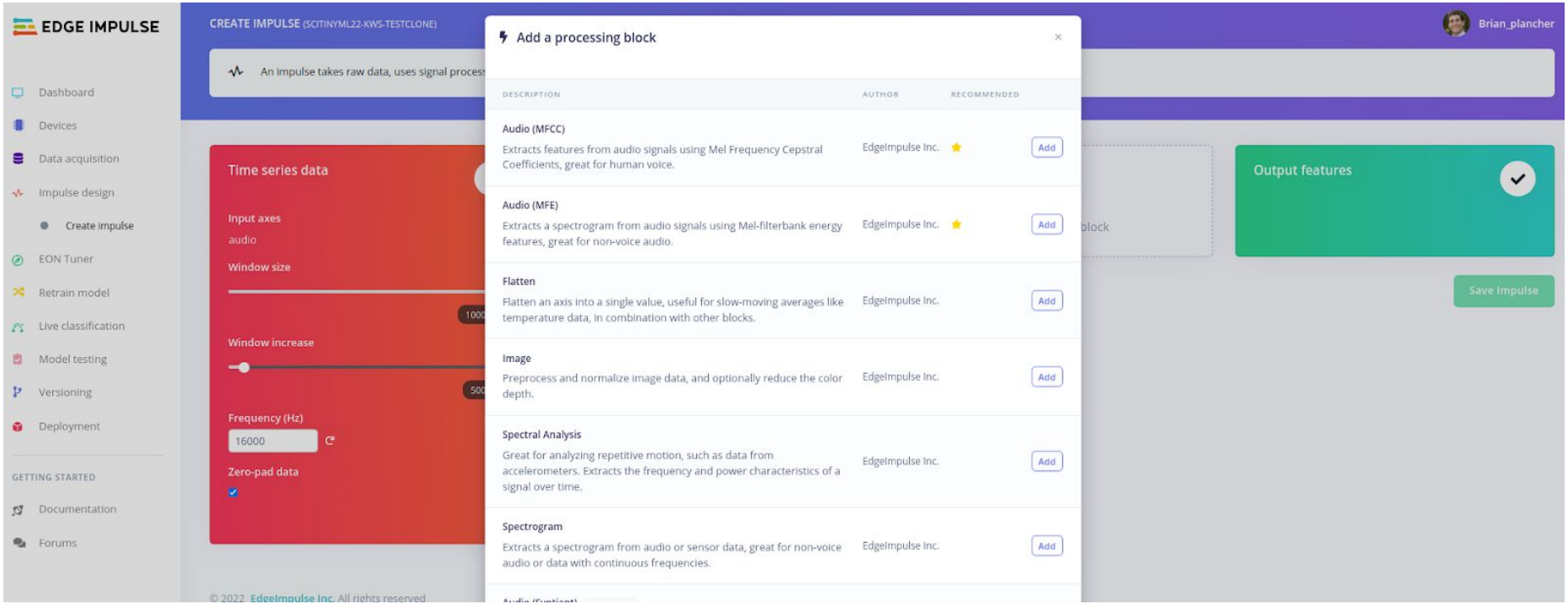
Edge Impulse Project Dashboard

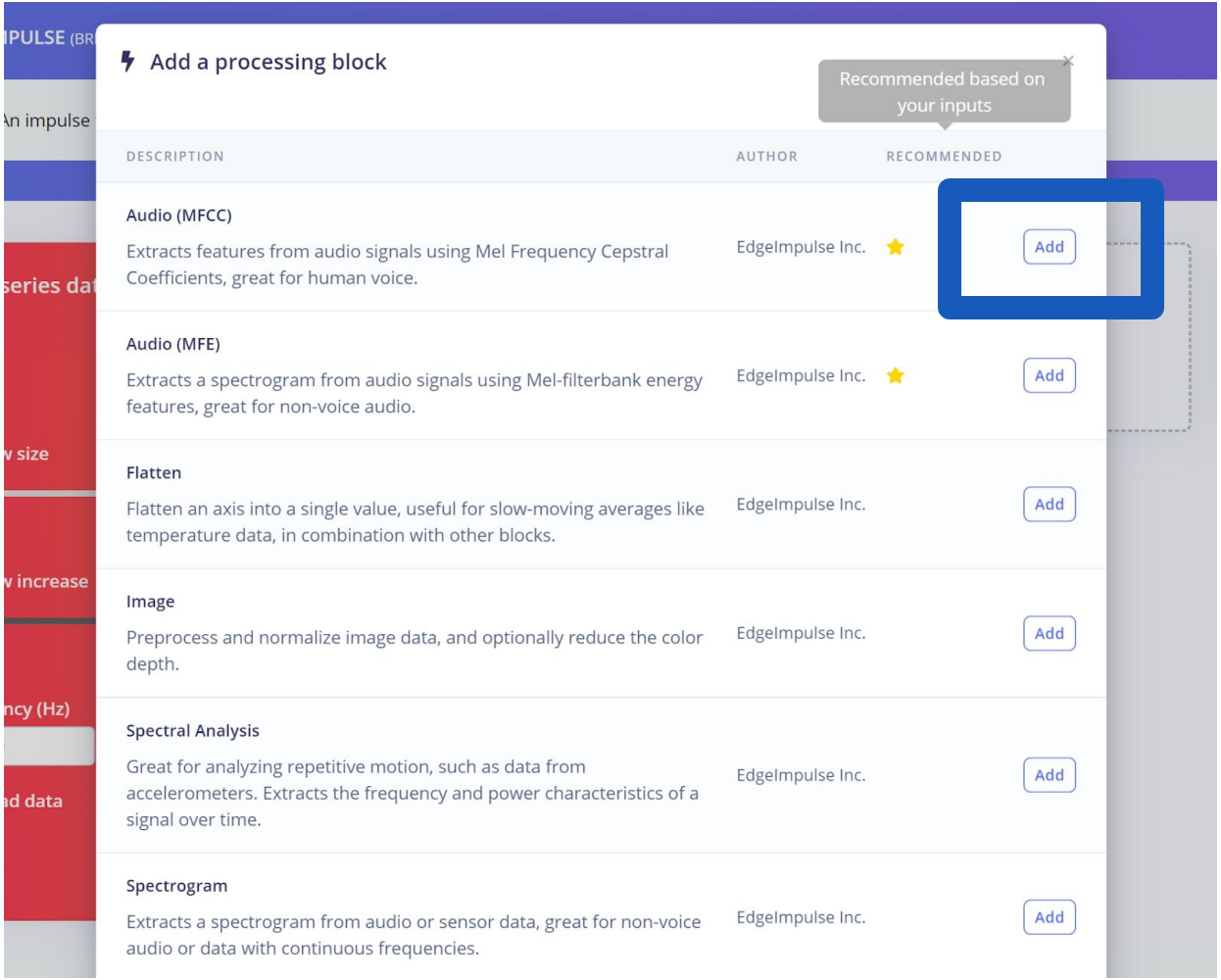


- Dashboard
- Devices
- Data acquisition
- Impulse design
  - Create impulse
- EON Tuner
- Retrain model
- Live classification
- Model testing
- Versioning
- Deployment





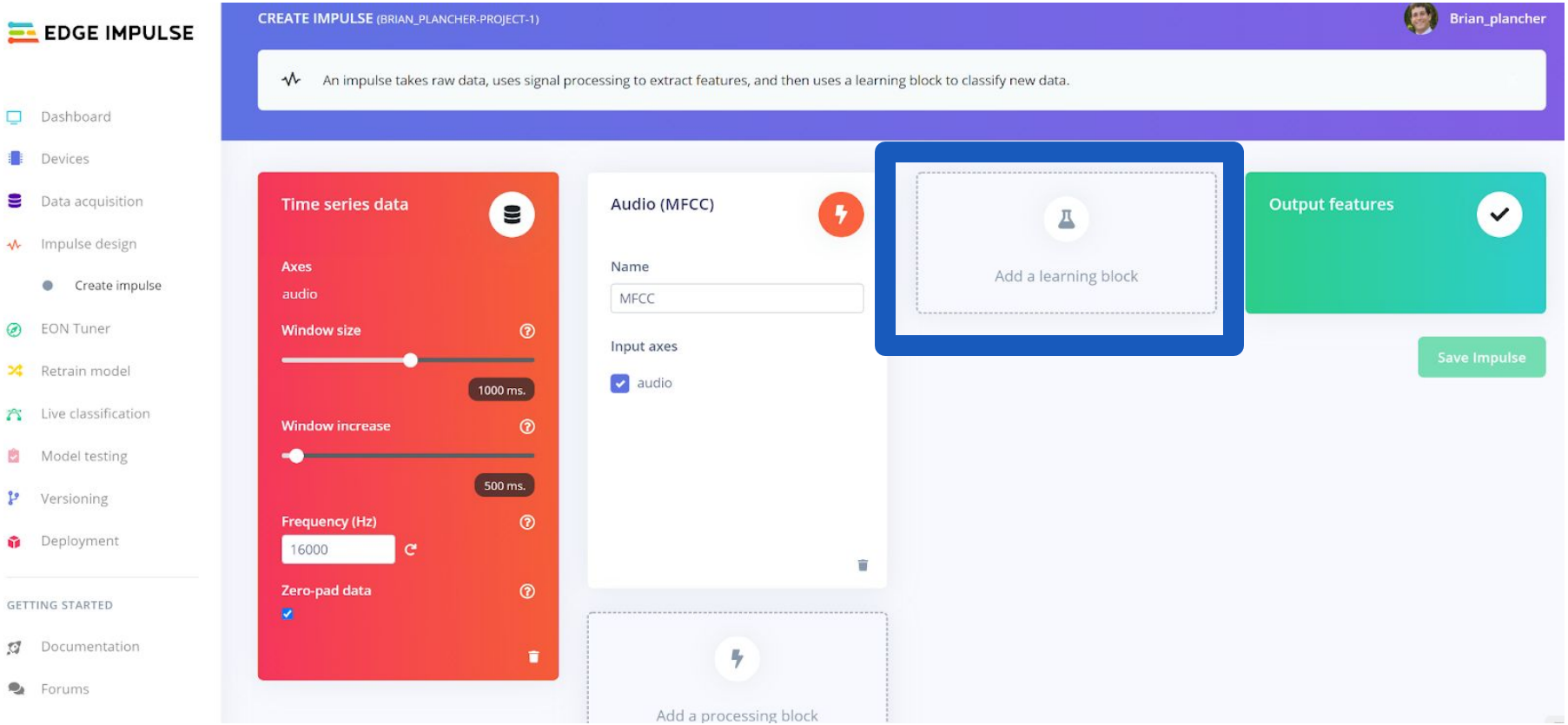


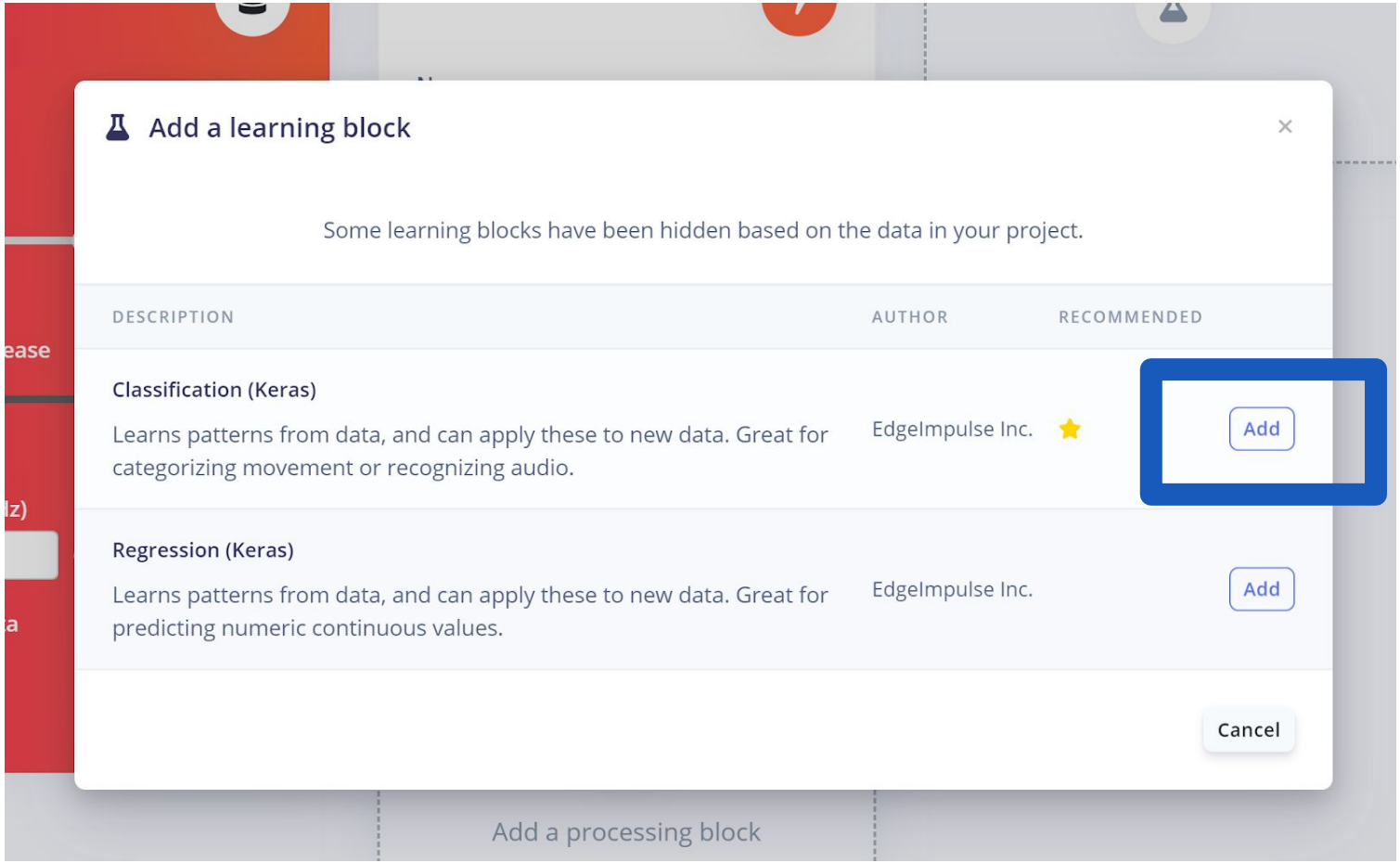


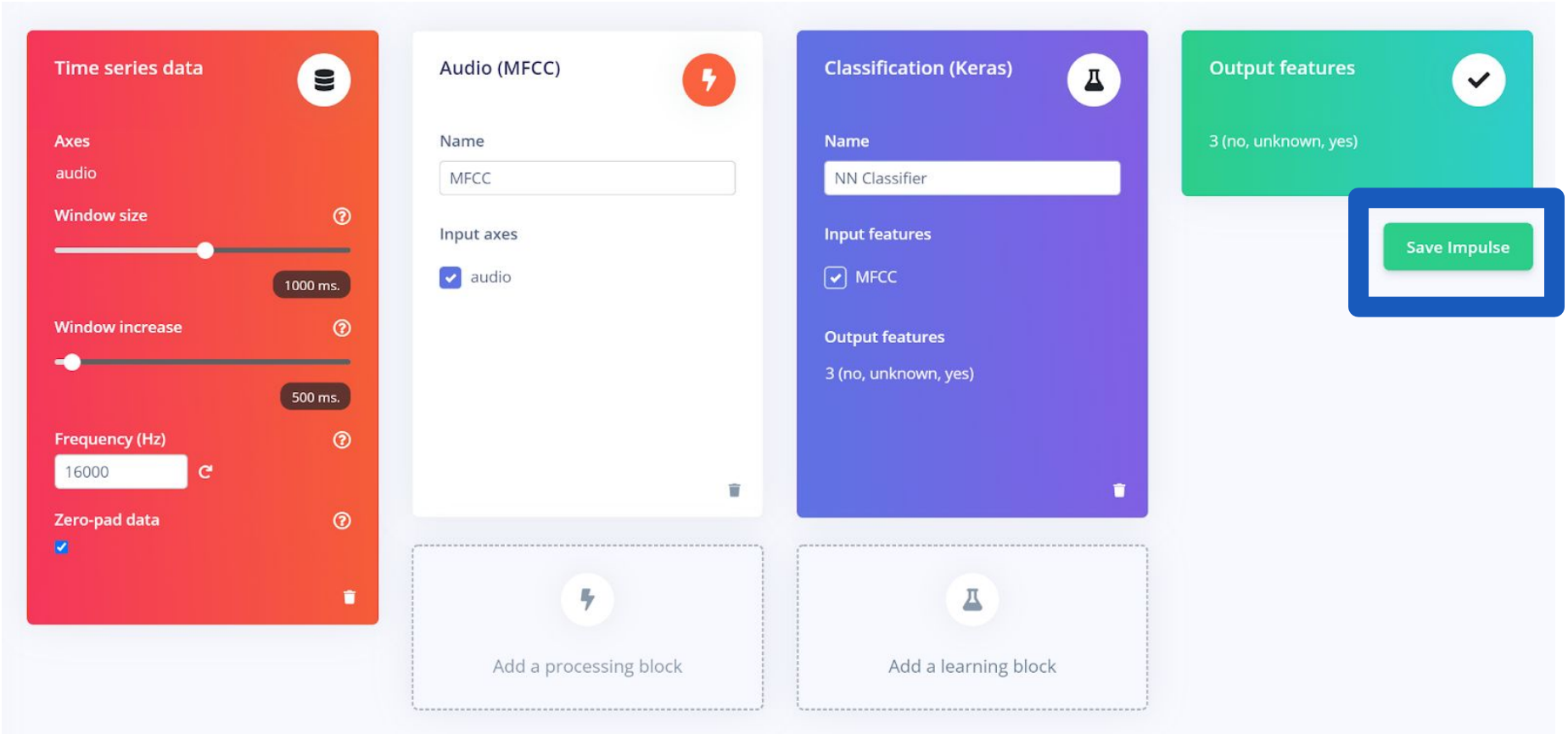
We'll keep things simple today and just add an MFCC but/and in future projects you can:

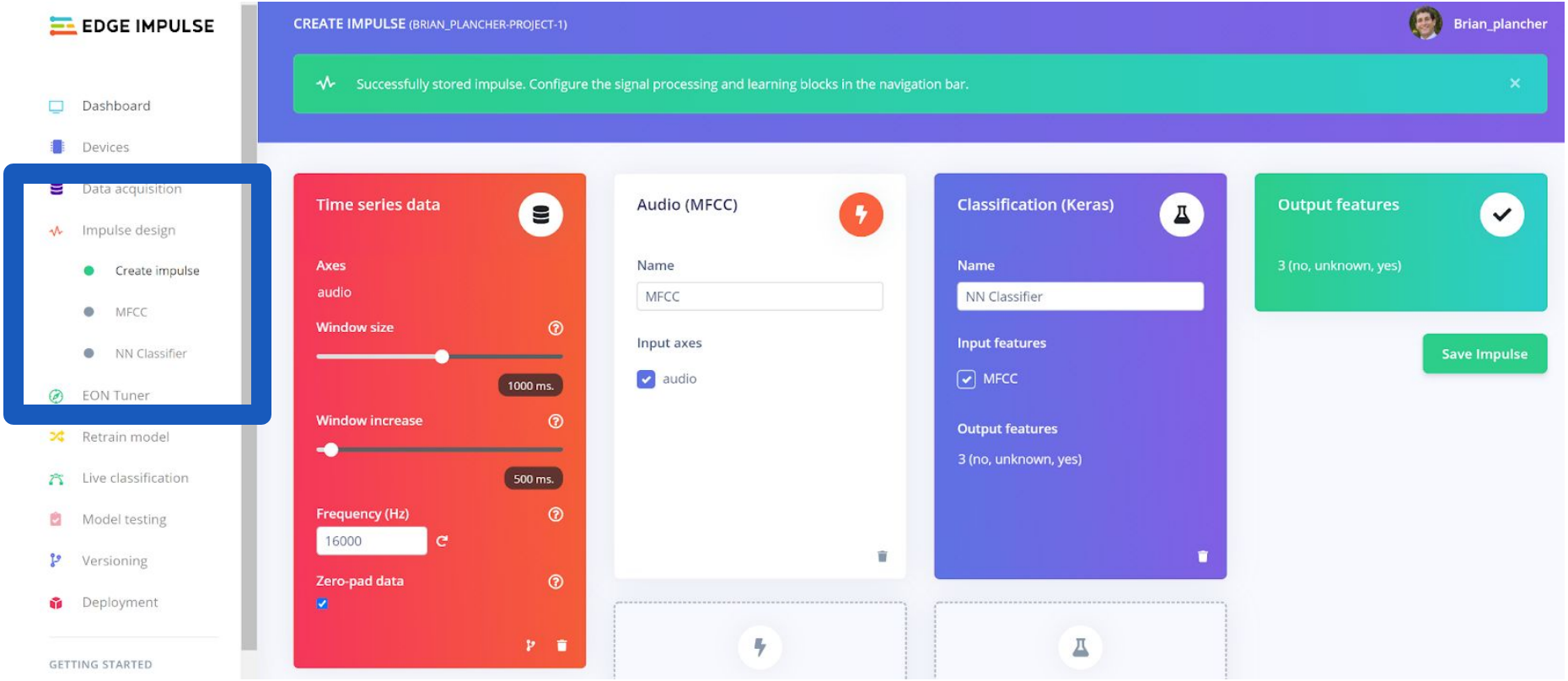
- **create your own blocks**
- **use multiple blocks**

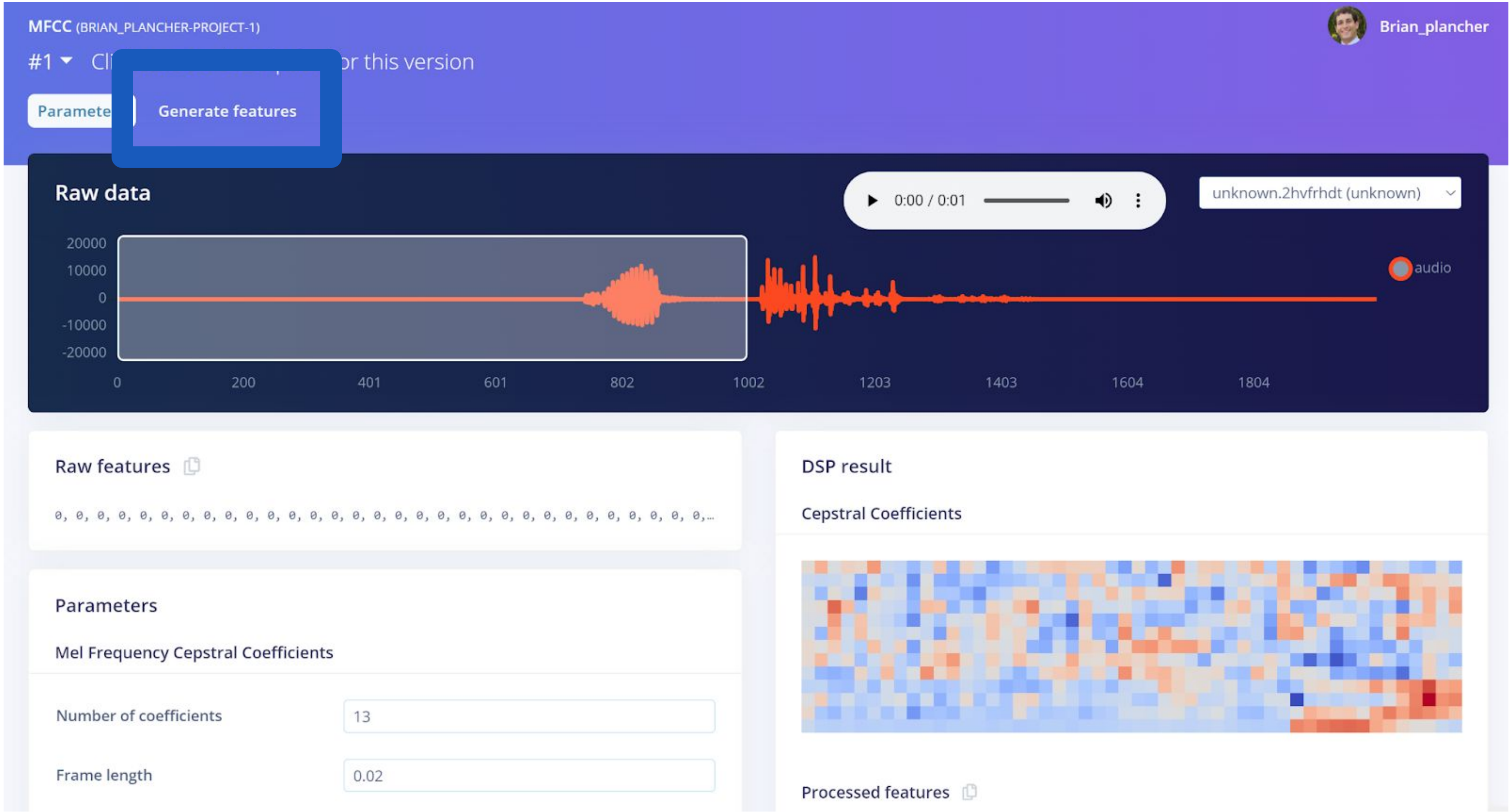
https://docs.edgeimpulse.com/docs/custom-blocks

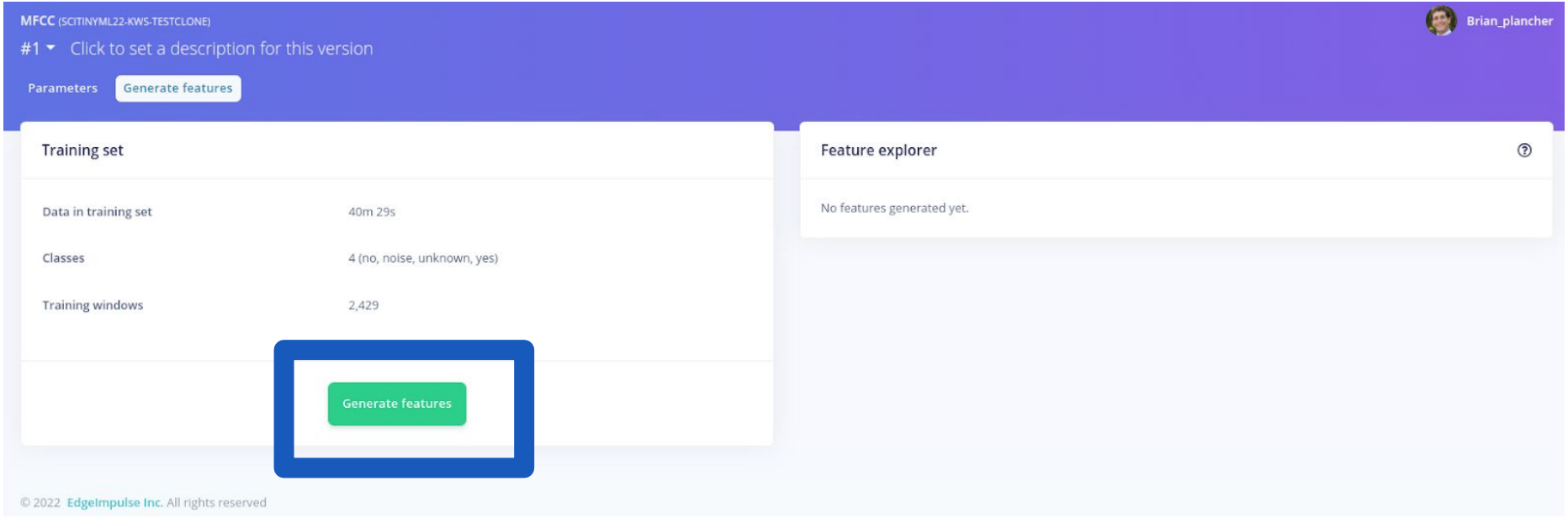


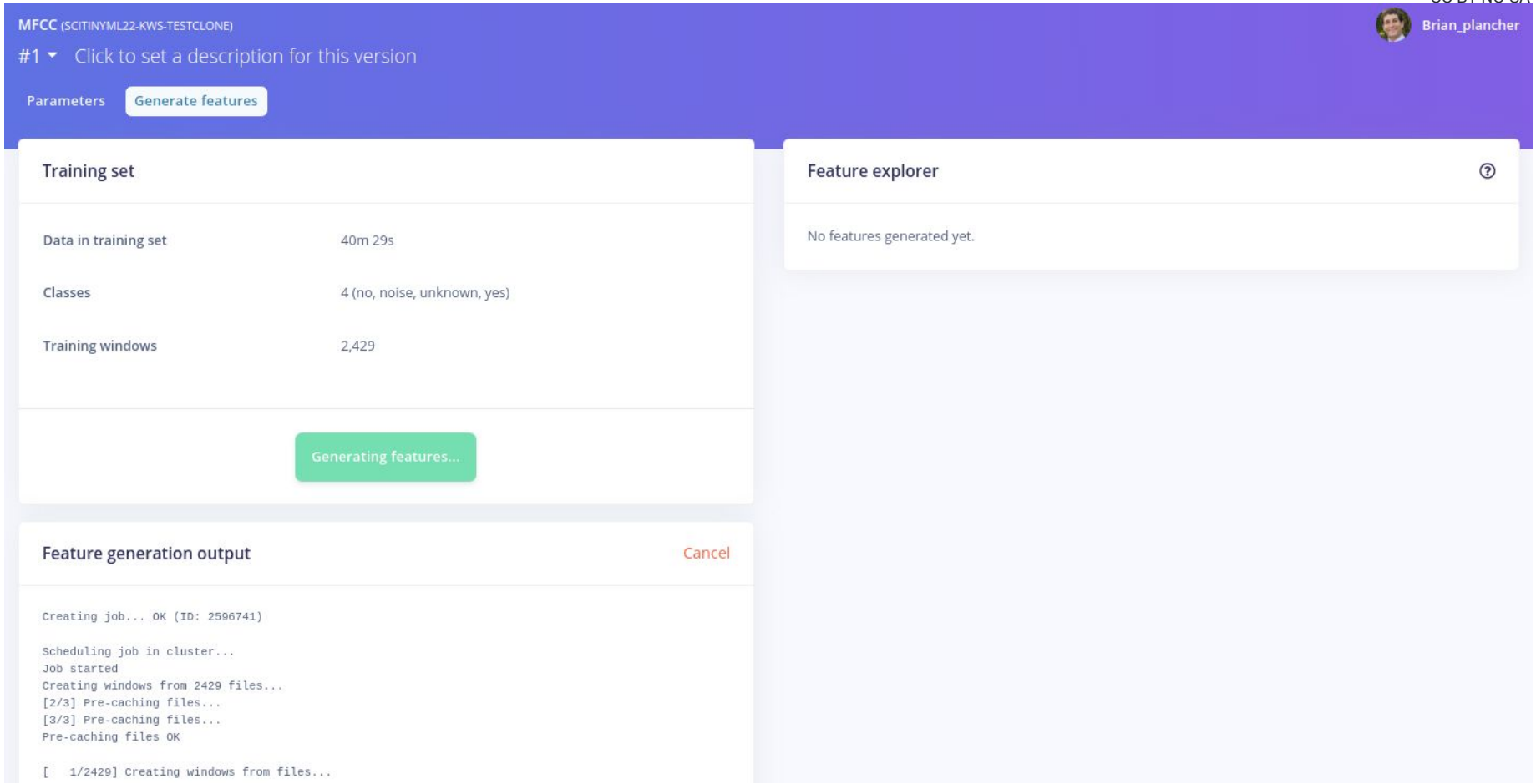




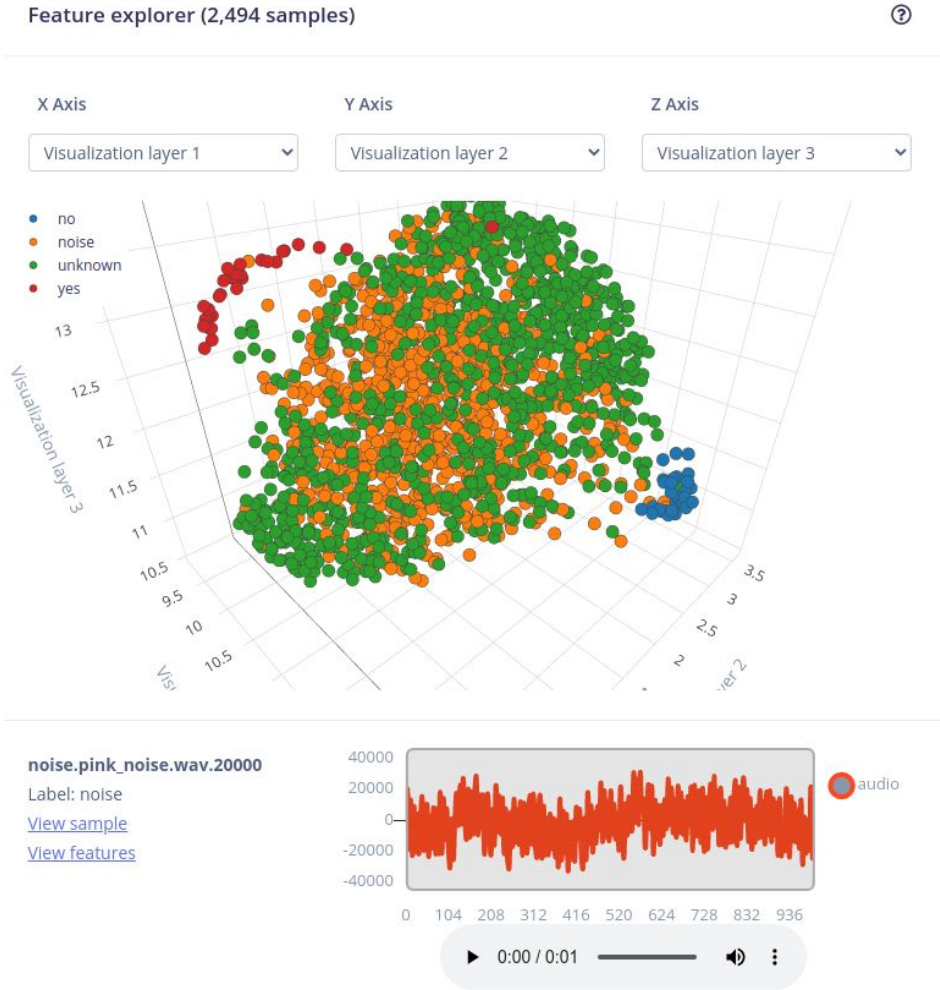


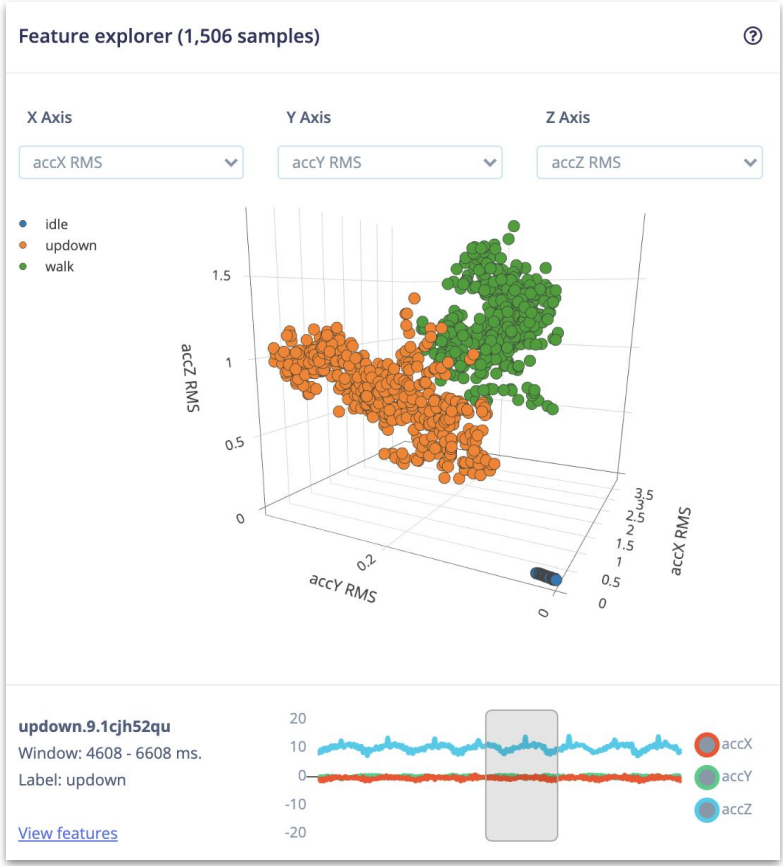






If you can visually see the clustering of the data then it is easier for the ML model to learn! (But its not required and provides no guarantees)





EDGE IMPULSE

MFCC (SCITINYML22-KWS-CLONE-V2)

#1 Click to set a description for this version

Brian_plancher

Parameters | Generate features

- Dashboard
- Devices
- Data acquisition
- Impulse design
  - Create impulse
  - NN Classifier
- Retrain model
- Live classification
- Model testing
- Versioning
- Deployment

GETTING STARTED

- Documentation
- Forums

Training set

Data in training set	41m 34s
Classes	4 (no, noise, unknown, yes)
Training windows	2,494

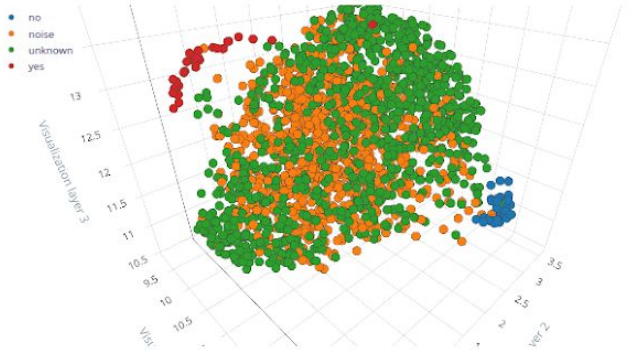
Generate features

Feature generation output

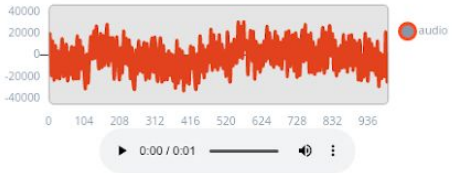
```
Still running...
        completed  150  /  500 epochs
        completed  200  /  500 epochs
Still running...
        completed  250  /  500 epochs
        completed  300  /  500 epochs
Still running...
        completed  350  /  500 epochs
        completed  400  /  500 epochs
Still running...
        completed  450  /  500 epochs
Wed Apr 27 19:18:09 2022 Finished embedding
Reducing dimensions for visualizations OK

Job completed
```

Feature explorer (2,494 samples)

X Axis: Visualization layer 1 | Y Axis: Visualization layer 2 | Z Axis: Visualization layer 3



noise.pink_noise.wav.20000
Label: noise
View sample
View features

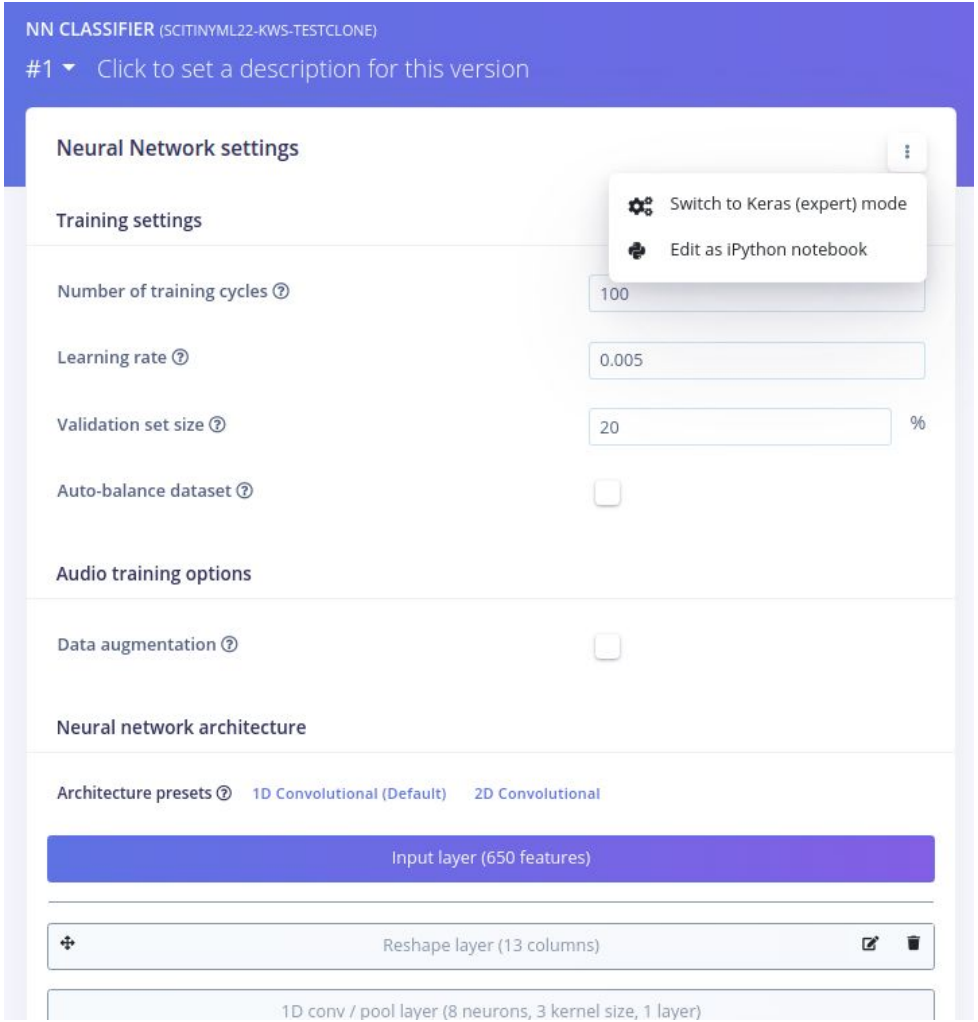


0:00 / 0:01

On-device performance

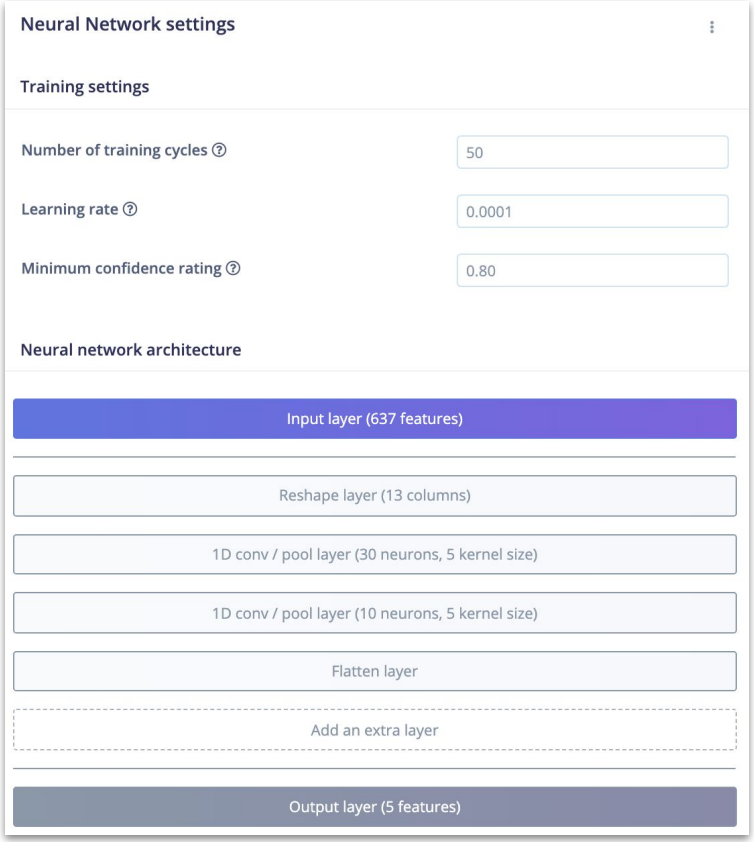
PROCESSING TIME
168 ms.

PEAK RAM USAGE
17 KB



Model Design with Edge Impulse

Pre-made neural network “blocks” that you can add!



Model Design with Edge Impulse

"**Expert**" mode to write your own TensorFlow code

Neural network architecture

Architecture presets (?) 1D Convolutional (Default) 2D Convolutional

- Input layer (650 features)
- Reshape layer (13 columns)
- 1D conv / pool layer (8 neurons, 3 kernel size, 1 layer)
- Dropout (rate 0.25)
- 1D conv / pool layer (16 neurons, 3 kernel size, 1 layer)
- Dropout (rate 0.25)
- Flatten layer
- Add an extra layer
- Output layer (3 features)

Start training

Neural network architecture

```
import tensorflow as tf
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense, InputLayer, Dropout, Conv1D, Conv2D,
    Flatten, Reshape, MaxPooling1D, MaxPooling2D, BatchNormalization,
    TimeDistributed
from tensorflow.keras.optimizers import Adam

# model architecture

model.add(Reshape((int(input_length / 13), 13), input_shape=(input_length, )))
model.add(Conv1D(8, kernel_size=3, activation='relu', padding='same'))
model.add(MaxPooling1D(pool_size=2, strides=2, padding='same'))

model.add(Conv1D(16, kernel_size=3, activation='relu', padding='same'))
model.add(MaxPooling1D(pool_size=2, strides=2, padding='same'))
model.add(Dropout(0.25))
model.add(Flatten())
model.add(Dense(classes, activation='softmax', name='y_pred'))

# this controls the learning rate
opt = Adam(lr=0.005, beta_1=0.9, beta_2=0.999)
# this controls the batch size, or you can manipulate the tf.data.Dataset objects
    yourself
BATCH_SIZE = 32
train_dataset = train_dataset.batch(BATCH_SIZE, drop_remainder=False)
validation_dataset = validation_dataset.batch(BATCH_SIZE, drop_remainder=False)
callbacks.append(BatchLoggerCallback(BATCH_SIZE, train_sample_count))

# train the neural network
model.compile(loss='categorical_crossentropy', optimizer=opt, metrics=['accuracy'])
model.fit(train_dataset, epochs=100, validation_data=validation_dataset, verbose=2,
    callbacks=callbacks)
```

Neural network architecture

Architecture presets ⓘ 1D Convolutional (Default) 2D Convolutional

Input layer (650 features)

Reshape layer (13 columns)

1D conv / pool layer (8 neurons, 3 kernel size, 1 layer)

Dropout (rate 0.25)

1D conv / pool layer (16 neurons, 3 kernel size, 1 layer)

Dropout (rate 0.25)

Flatten layer

Add an extra layer

Output layer (3 features)

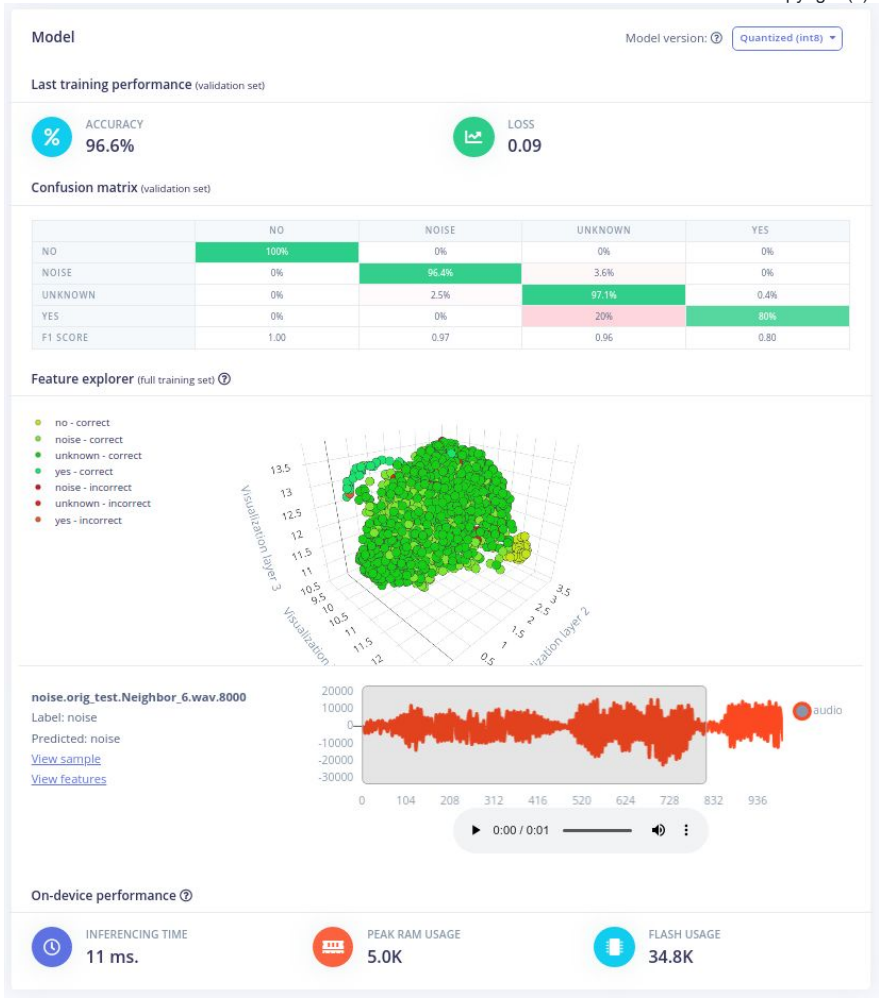
Start training

Neural network architecture

```python
import tensorflow as tf
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense, InputLayer, Dropout, Conv1D, Conv2D,
    Flatten, Reshape, MaxPooling1D, MaxPooling2D, BatchNormalization,
    TimeDistributed
from tensorflow.keras.optimizers import Adam

# model architecture
model = Sequential()
model.add(Reshape((int(input_length / 13), 13), input_shape=(input_length, )))
model.add(Conv1D(8, kernel_size=3, activation='relu', padding='same'))
model.add(MaxPooling1D(pool_size=2, strides=2, padding='same'))
model.add(Dropout(0.25))
model.add(Conv1D(16, kernel_size=3, activation='relu', padding='same'))
model.add(MaxPooling1D(pool_size=2, strides=2, padding='same'))
model.add(Dropout(0.25))
model.add(Flatten())
model.add(Dense(classes, activation='softmax', name='y_pred'))

# this controls the learning rate
opt = Adam(lr=0.005, beta_1=0.9, beta_2=0.999)
# this controls the batch size, or you can manipulate the tf.data.Dataset objects
    yourself
BATCH_SIZE = 32
train_dataset = train_dataset.batch(BATCH_SIZE, drop_remainder=False)
validation_dataset = validation_dataset.batch(BATCH_SIZE, drop_remainder=False)
```

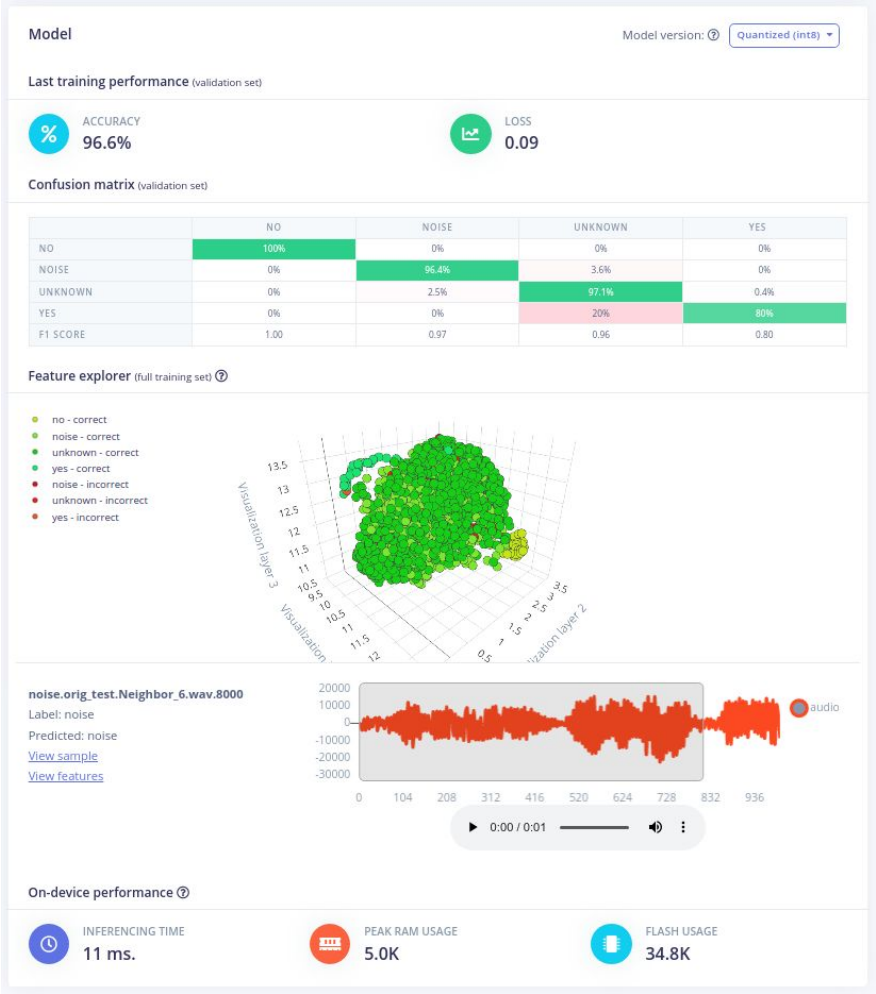
For now just stick with the defaults but/and you can easily design **any model** you want and use **any optimizer** you want using **TensorFlow**!

Final Accuracy



Final Accuracy

Accuracy Breakdown



Confusion Matrix

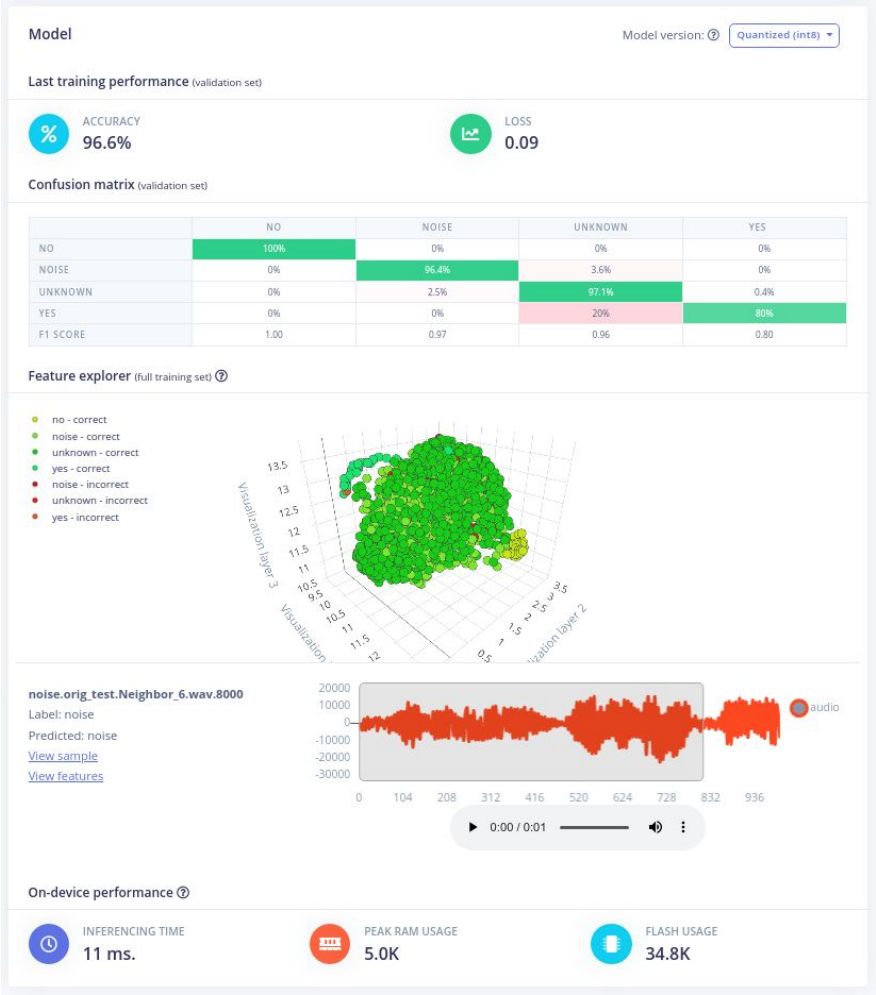
	Actual Output = Yes	Actual Output = No
Predicted Output = Yes	**# of True Positive**	# of False Positive ***Type 1 Error***
Predicted Output = No	# of False Negative ***Type 2 Error***	**# of True Negative**

Final Accuracy

Accuracy Breakdown

Feature Explorer

Individual Data Points



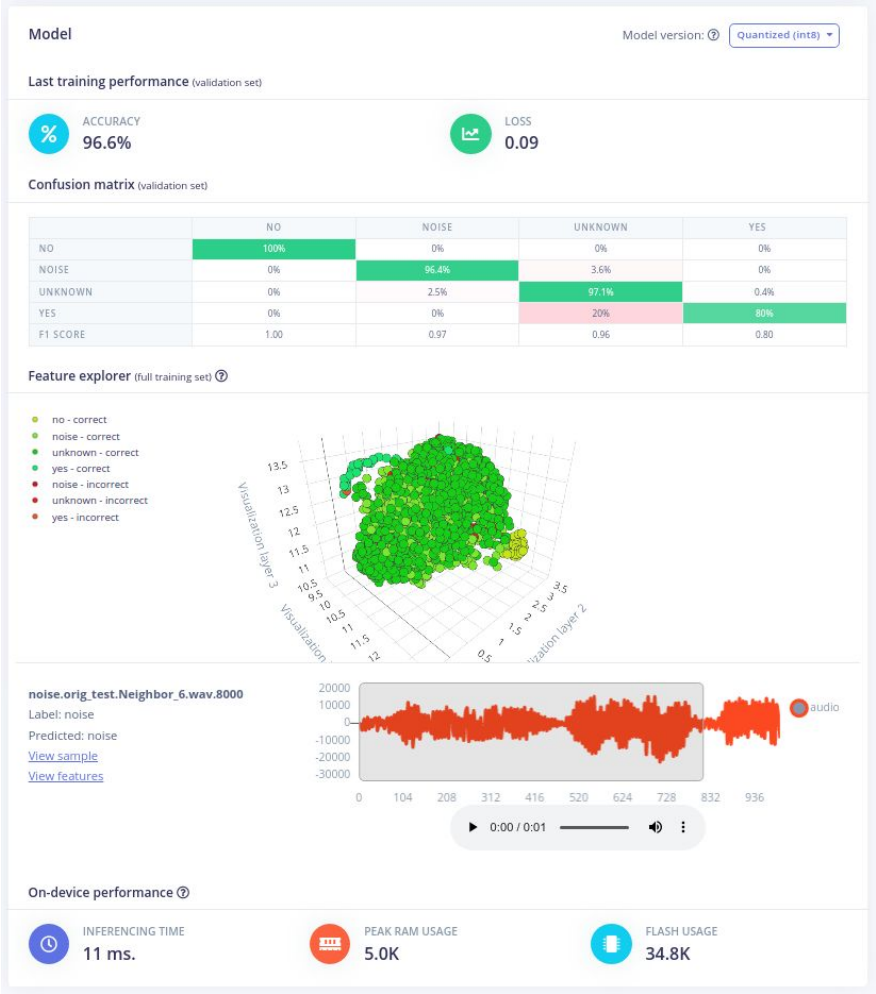
Final Accuracy

Accuracy Breakdown

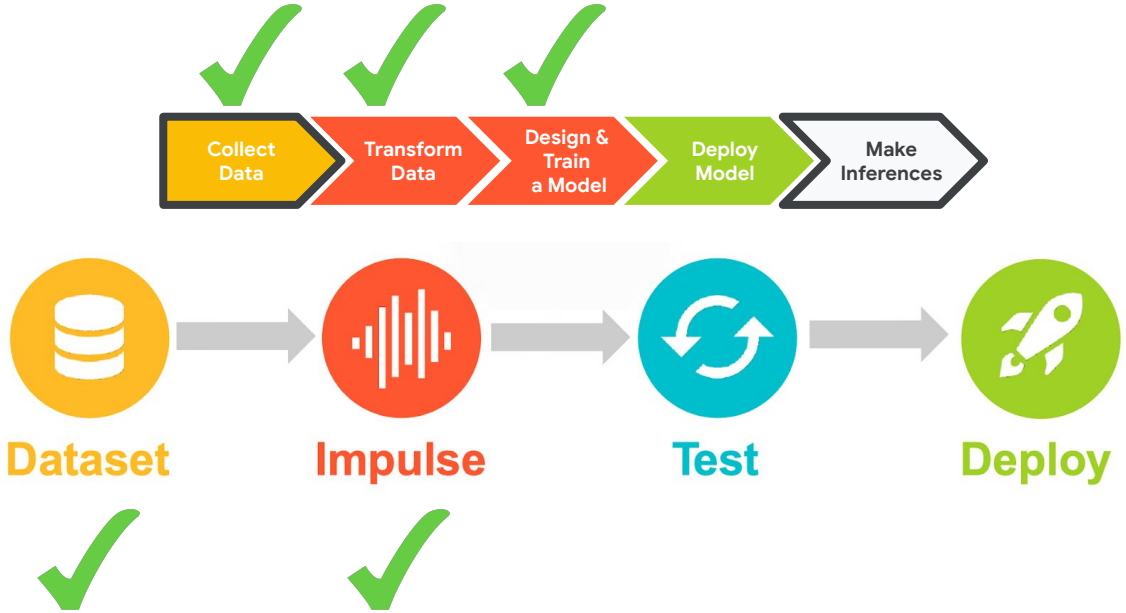
Feature Explorer

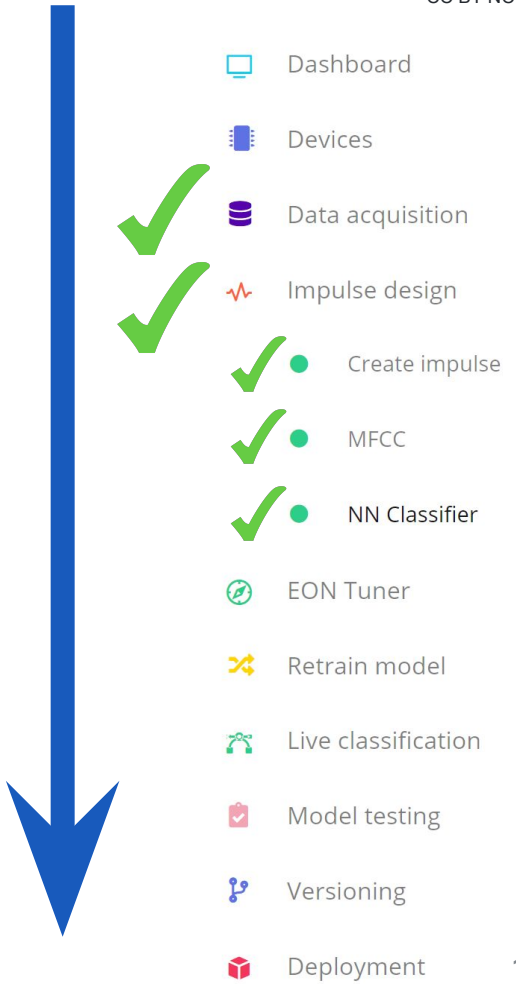
Individual Data Points

Expected runtime/memory



Edge Impulse Project Dashboard





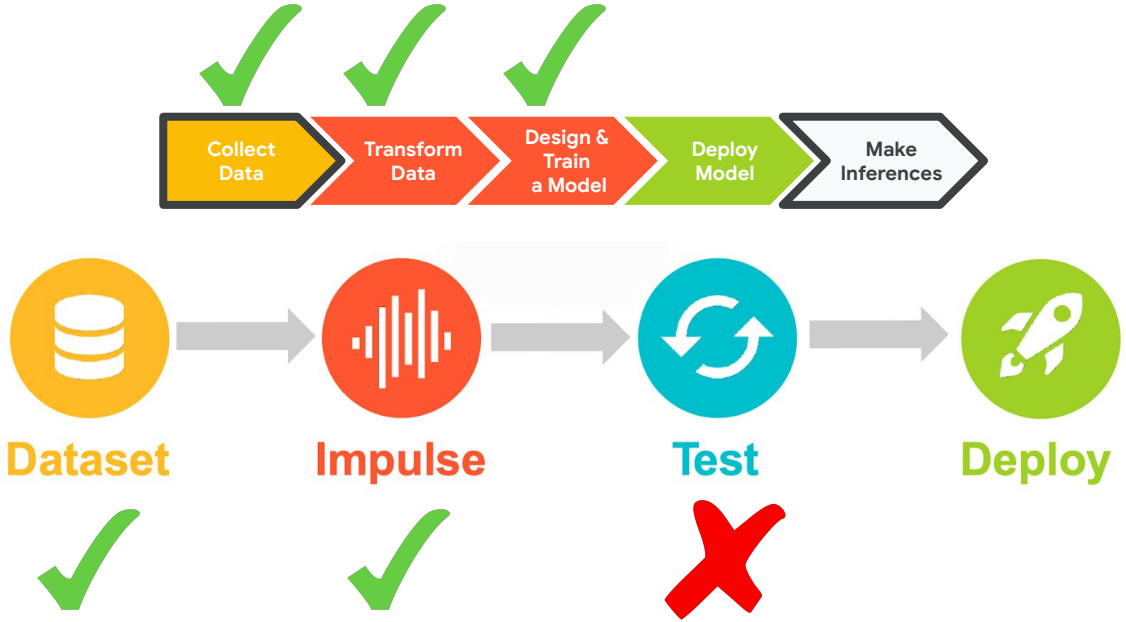
Today's Agenda

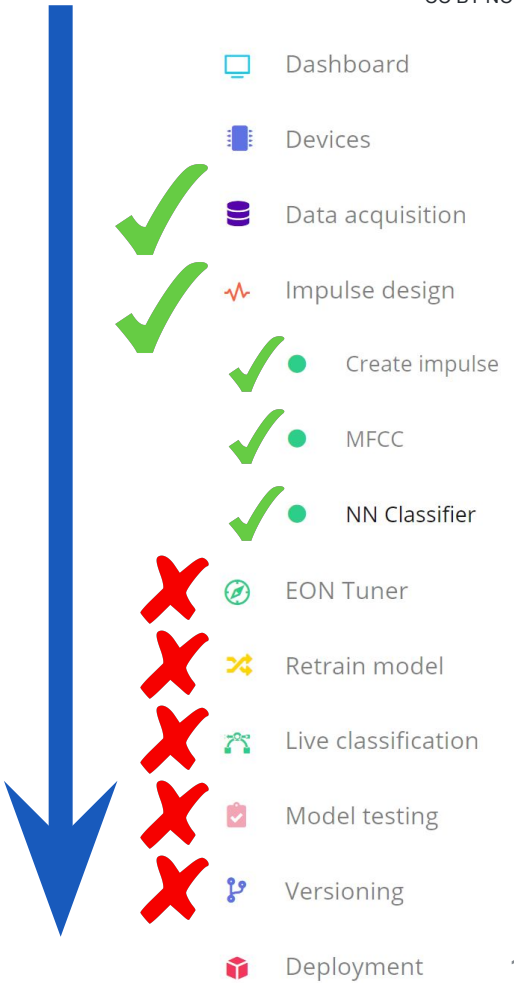
- Preprocessing for Keyword Spotting
- Convolutional Neural Networks for Image Classification
- Hands-on: KWS Data Collection with Edge Impulse
- **Hands-on: Training our Model with Edge Impulse**
- Hands-on: Testing our Model in the Real World
- Summary

Today's Agenda

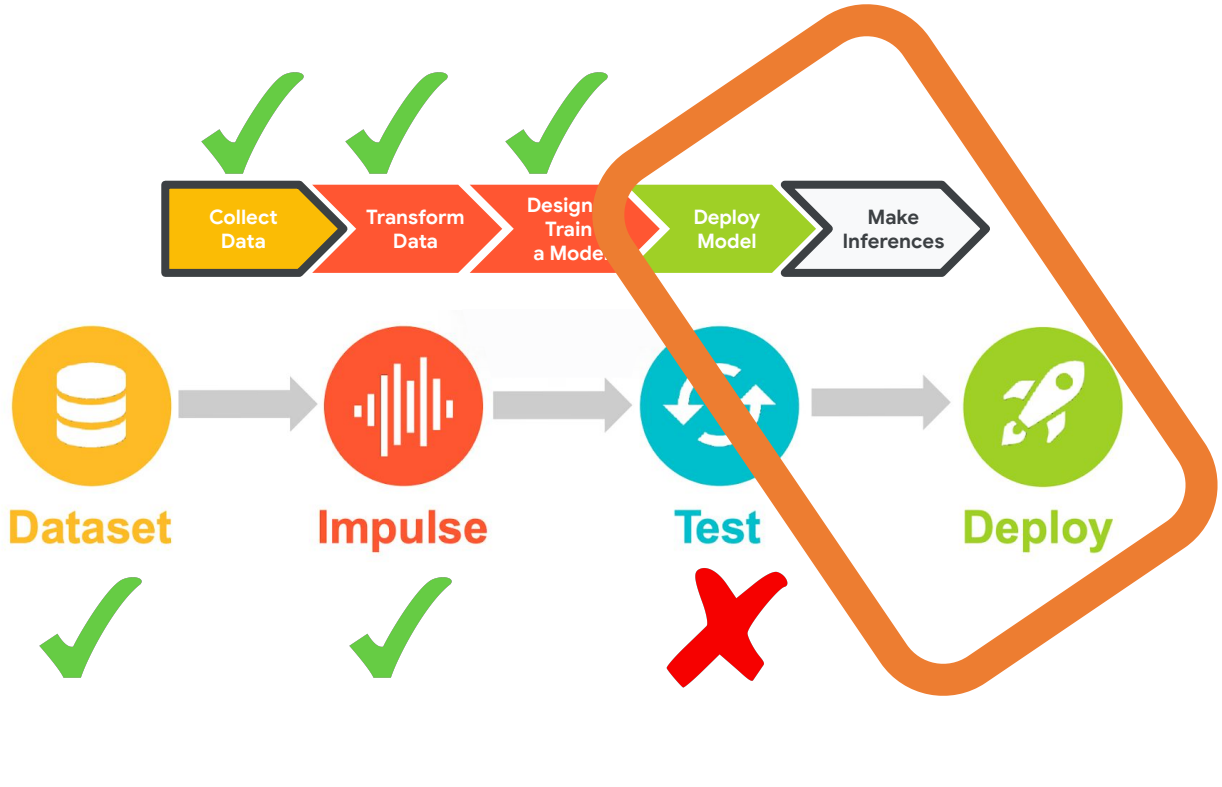
- Preprocessing for Keyword Spotting
- Convolutional Neural Networks for Image Classification
- Hands-on: KWS Data Collection with Edge Impulse
- Hands-on: Training our Model with Edge Impulse
- **Hands-on: Testing our Model in the Real World**
- Summary

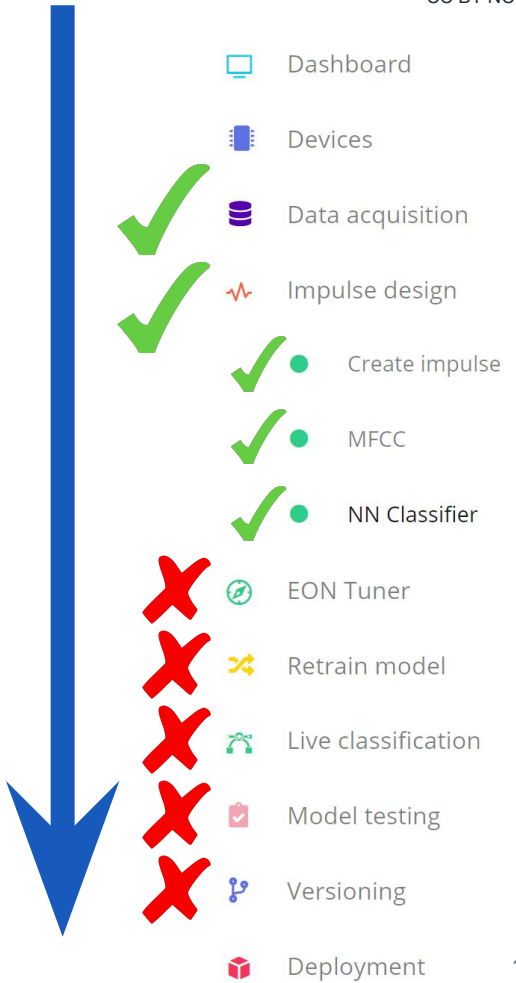
Edge Impulse Project Dashboard

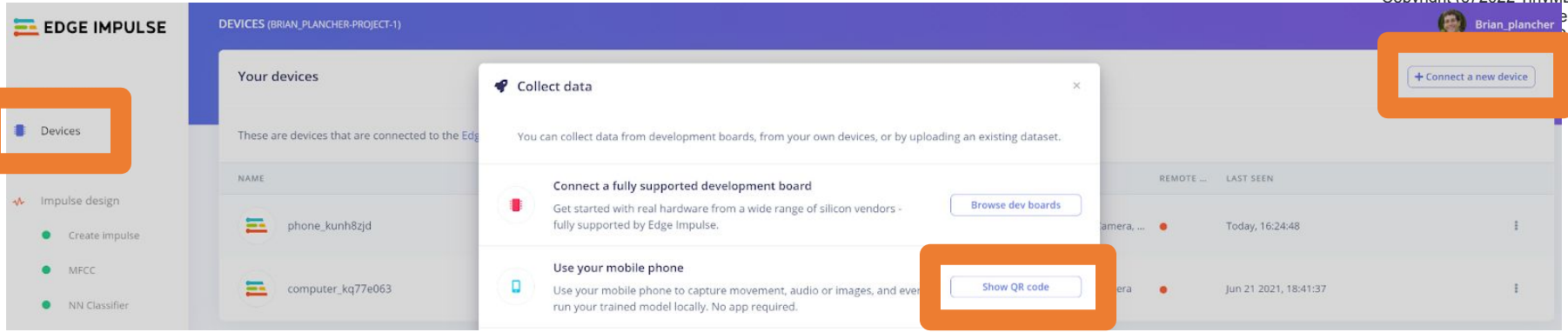


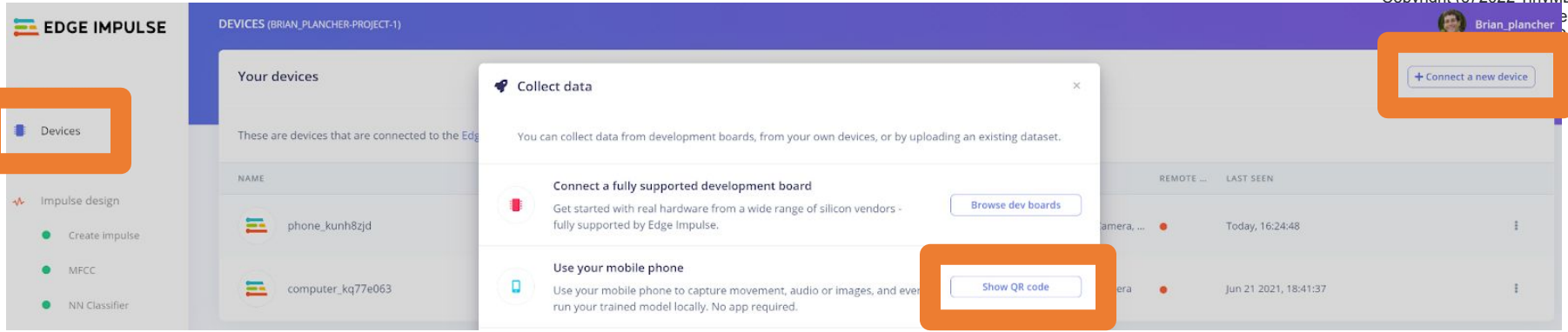


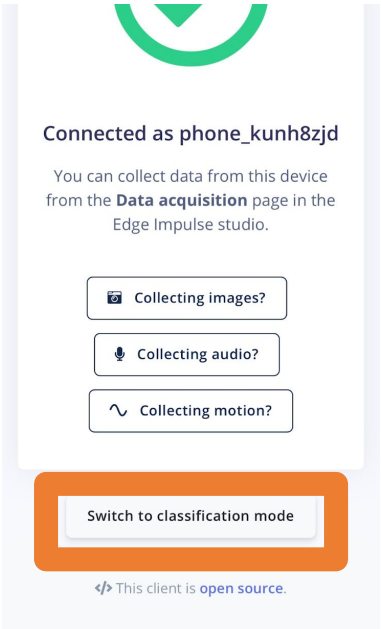
Edge Impulse Project Dashboard

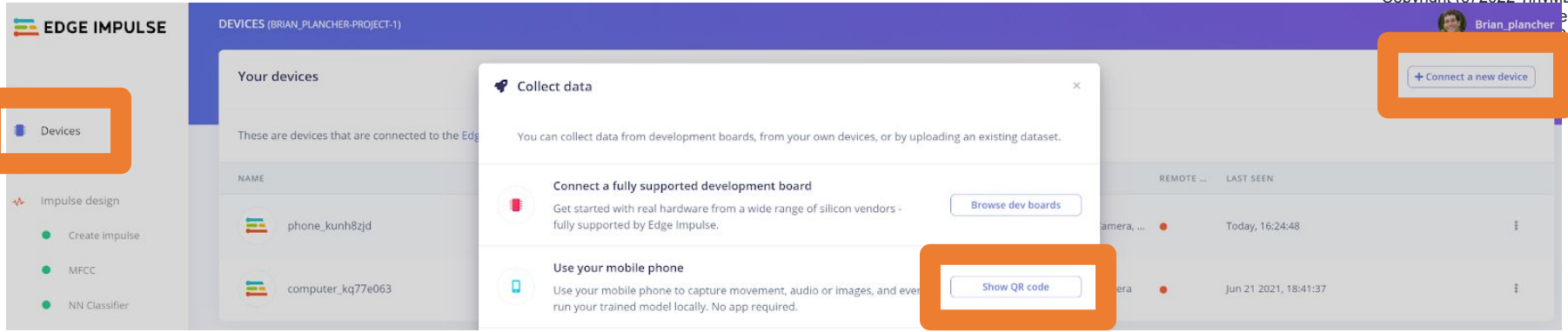


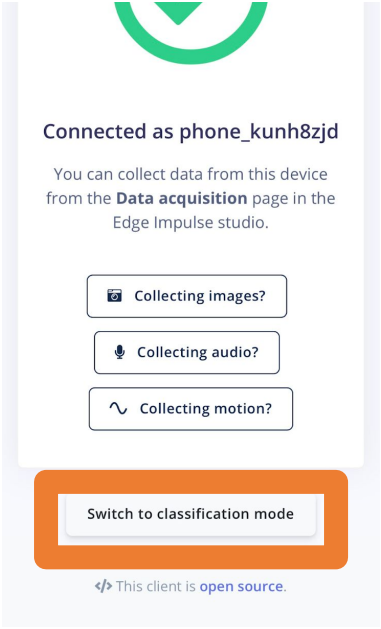


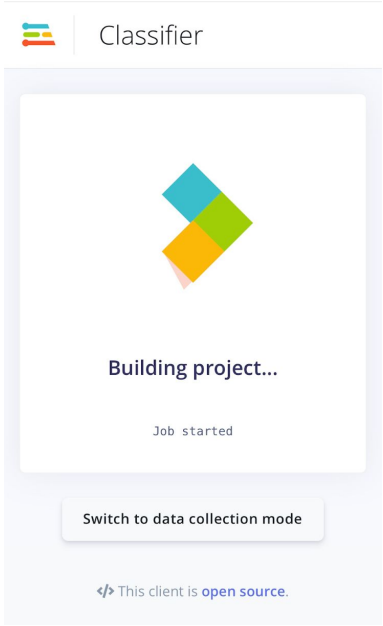


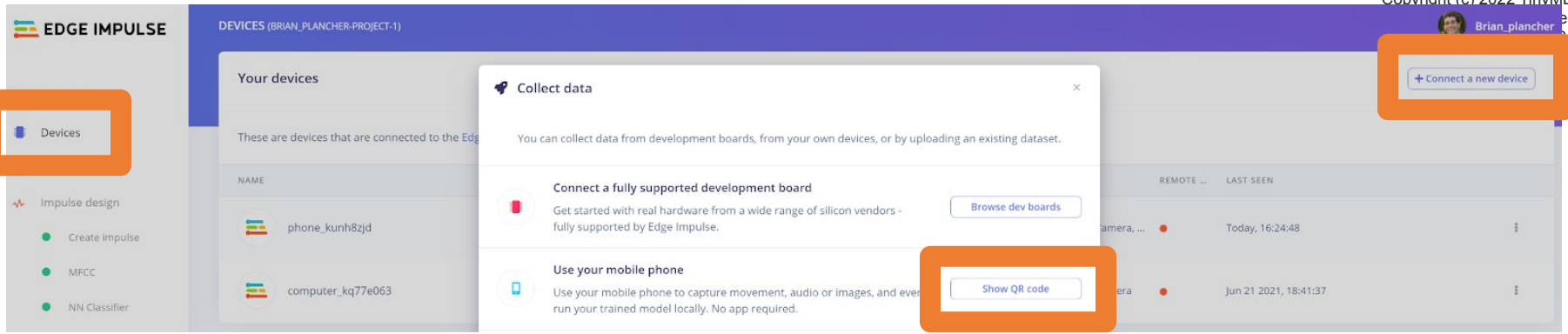


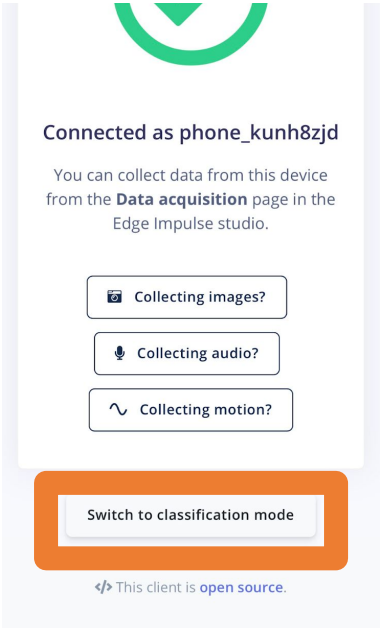


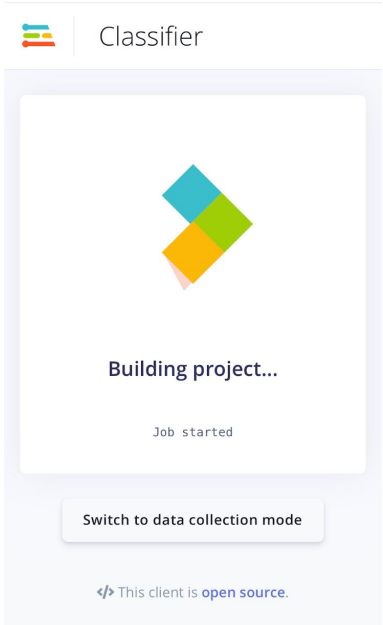


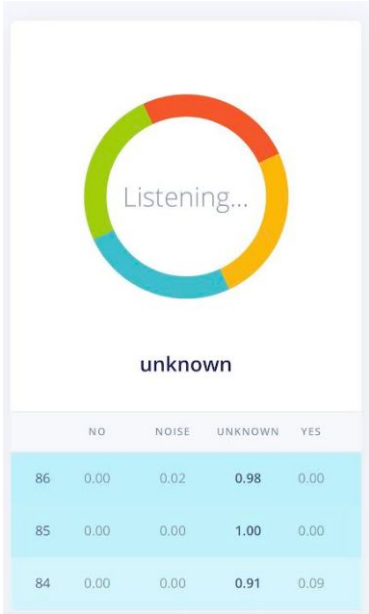






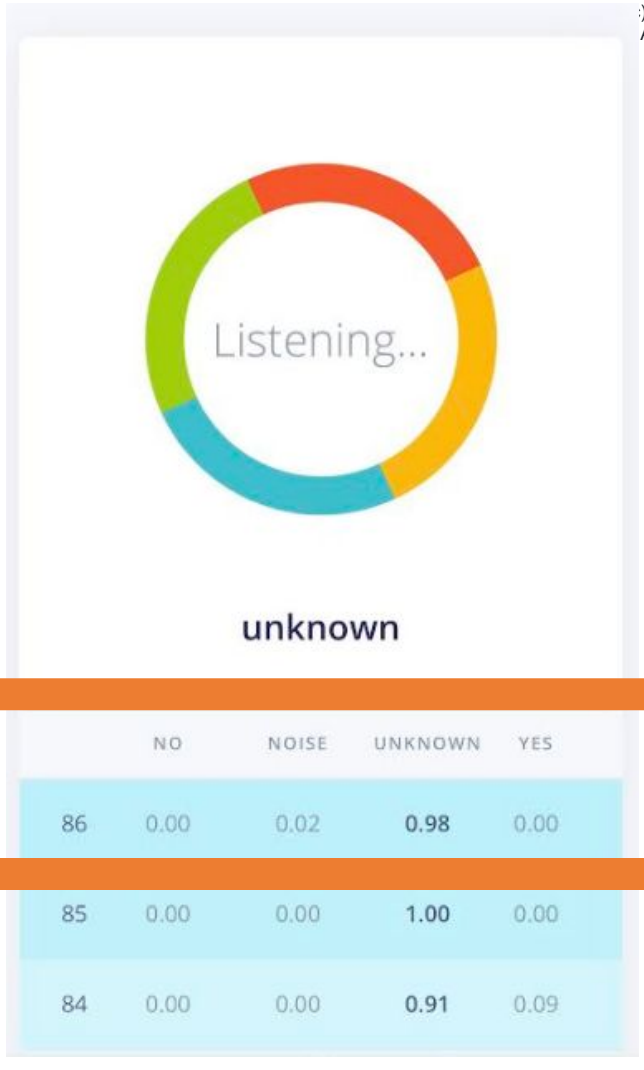






Deploy and Test your Model

Shows the **score** for **(confidence that the current sounds is)** each of the various keywords and unknown and bolds the highest score.



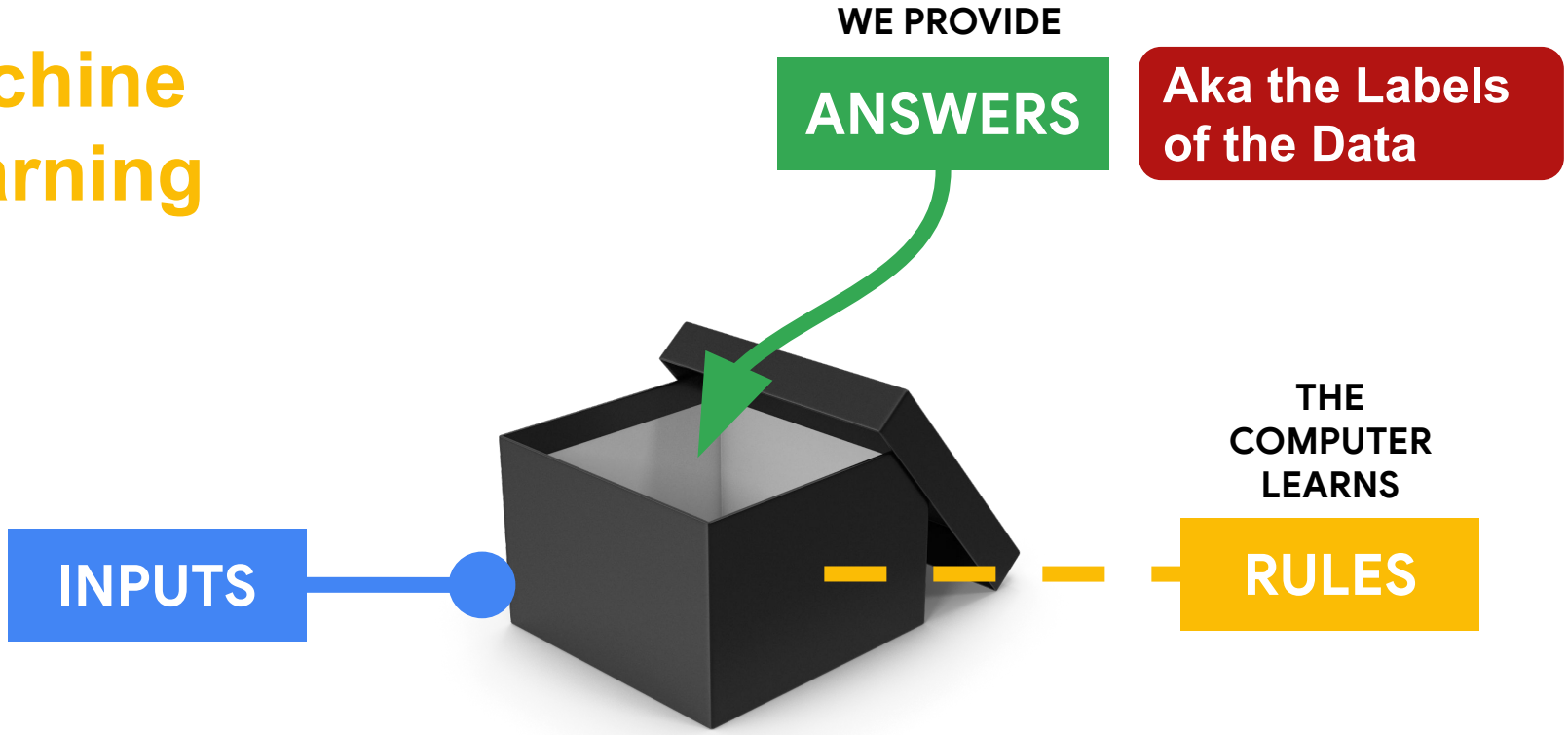
Today's Agenda

- Preprocessing for Keyword Spotting
- Convolutional Neural Networks for Image Classification
- Hands-on: KWS Data Collection with Edge Impulse
- Hands-on: Training our Model with Edge Impulse
- **Hands-on: Testing our Model in the Real World**
- Summary

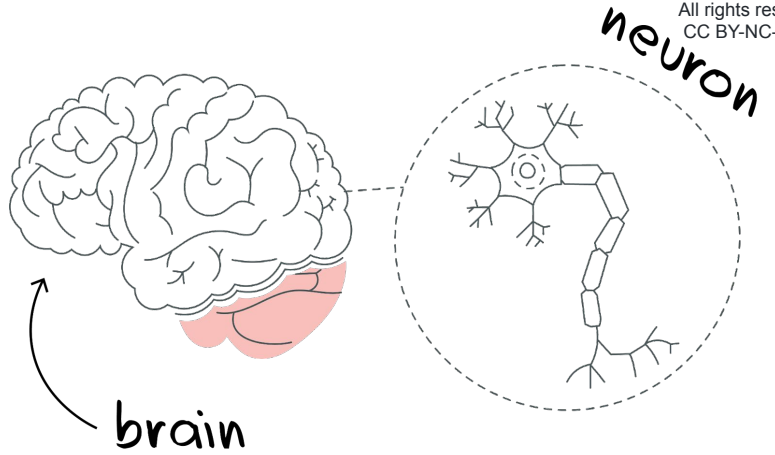
Today's Agenda

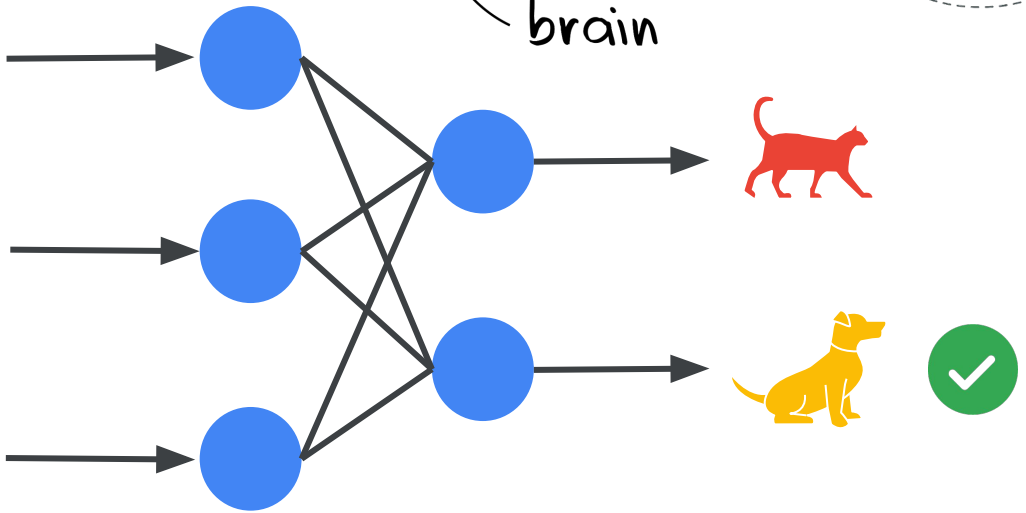
- Preprocessing for Keyword Spotting
- Convolutional Neural Networks for Image Classification
- Hands-on: KWS Data Collection with Edge Impulse
- Hands-on: Training our Model with Edge Impulse
- Hands-on: Testing our Model in the Real World
- **Summary**

Machine Learning



Deep Learning with Neural Networks





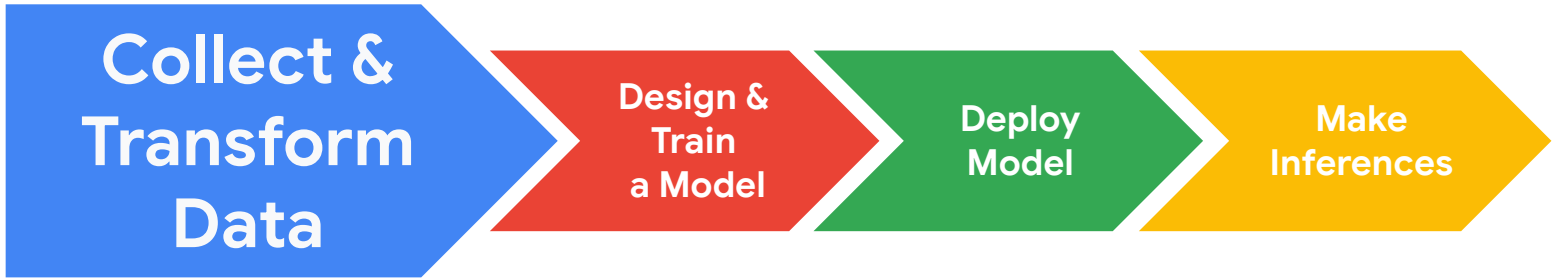
Colab Link

Features can be found with Convolutions

-1	0	1
-2	0	2
-1	0	1

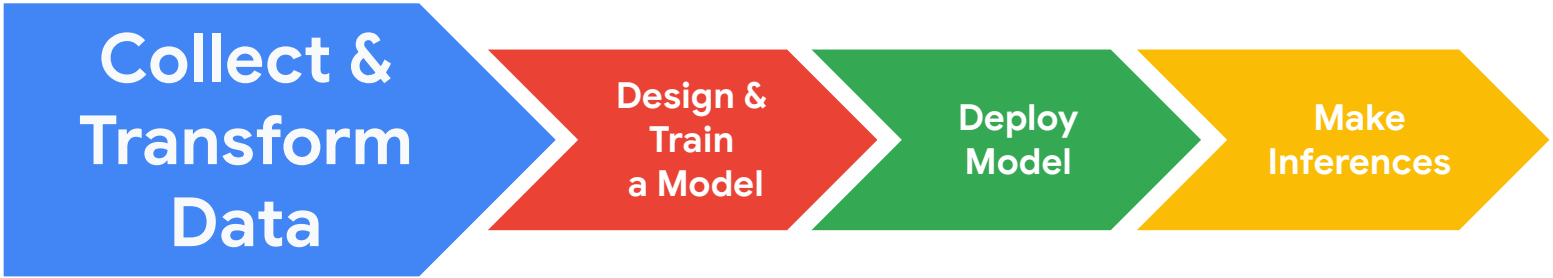
Features

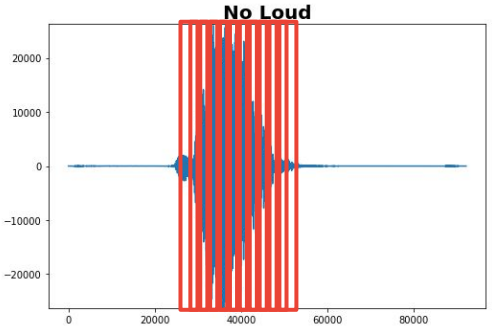
The (Tiny) Machine Learning **Workflow**

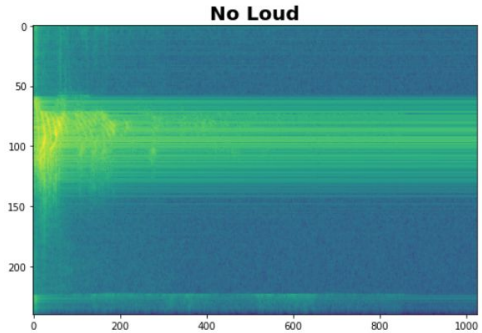


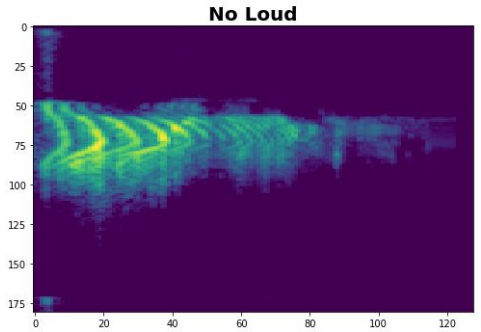
If ML is going to be everywhere we need to consider how to best collect GOOD data RESPONSIBLY

The (Tiny) Machine Learning **Workflow**

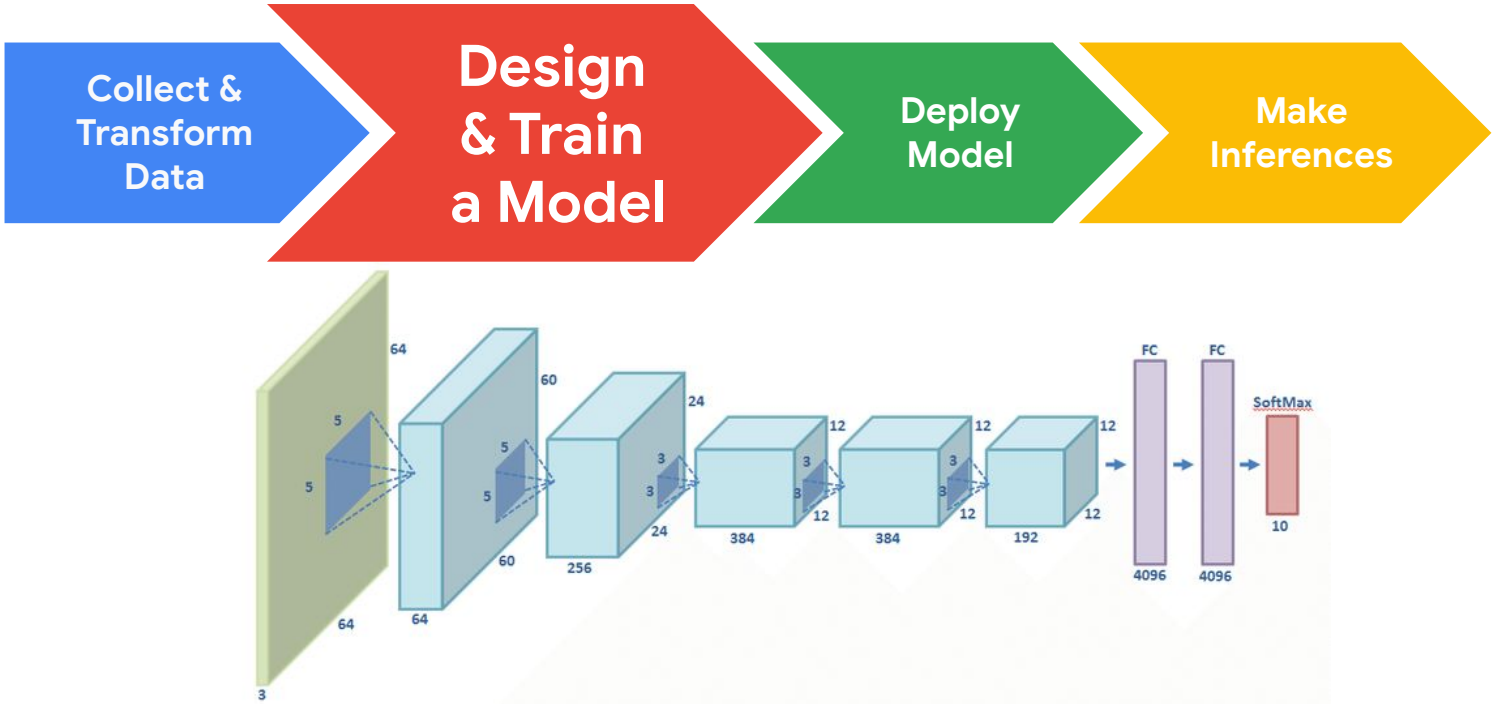




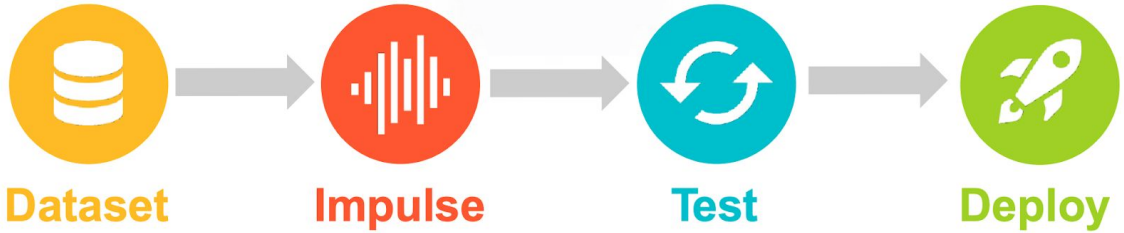




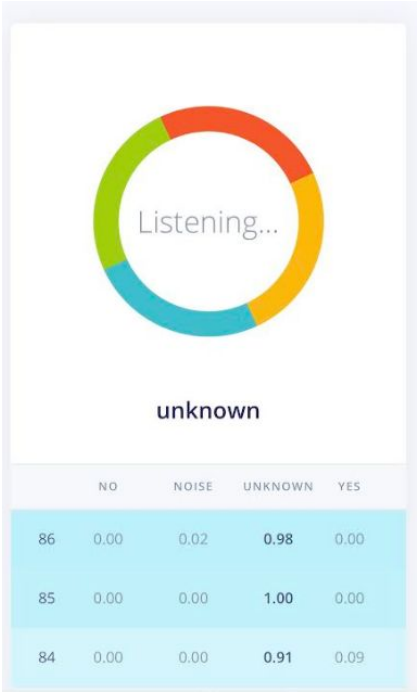
The (Tiny) Machine Learning **Workflow**



The (Tiny) Machine Learning **Workflow**



Edge Impulse Simplifies Training and Deployment



Better Data = Better Models!

hágoónee’ 👋

see you **again at 12pm (Mountain Time)**